Evaluation of Pipelined Switch Architectures for ATM Networks

by

Mohammed Kaleemuddin

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER ENGINEERING

June, 1997

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

EVALUATION OF PIPELINED SWITCH ARCHITECTURES FOR ATM NETWORKS

BY

MOHAMMED KALEEMUDDIN

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER ENGINEERING

JUNE 1997

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

DHAHRAN, SAUDI ARABIA

COLLEGE OF GRADUATE STUDIES

This thesis, written by

MOHAMMAD KALEEMUDDIN

under the direction of his Thesis Advisor and approved by his Thesis Committee, has been presented to and accepted by the Dean of the College of Graduate Studies, in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER ENGINEERING

Thesis Committee

Dr. Mayez Al – Mouhamed (Chairman)

Dr. Habib Youssef (Co – Chairman)

Dr. Hasan R.Barada (Member)

Dr. Khalid M. Al – Tawil
(Department Chairman)

Dr. Abdallah M. Al – Shehri
(Dean, College of Graduate Studies)

Date

Dedicated as a humble tribute to

my beloved mother

whose prayers, sacrifice, inspiration and love

led to this accomplishment

and to

my loving father

Acknowledgements

In the name of Allah, Most Gracious, Most Merciful. Read in the name of thy Lord and Cherisher, Who created. Created man from a {leech-like } *clot. Read and thy Lord is Most Bountiful. He Who taught* {*the use of*} *the pen. Taught man that which he knew not. Nay, but man doth transgress all bounds. In that he looketh upon himself as self-sufficient. Verily, to thy Lord is the return* {*of all*}.

(The Holy Quran, Surah 96)

All praises are for ALLAH *subhanahu-wa-ta-Aaala*, the Most Compassionate, the Most Merciful. May peace and blessings be upon Prophet Muhammad, and his family. I thank Almighty Allah for giving me the knowledge and patience to complete this work. May He guide me and the whole humanity to the right path *(Aameen)*.

I acknowledge the support and facilities provided by the King Fahd University of Petroleum and Minerals for this work.

I would like to express my profound gratitude and appreciation to my thesis committee chairman Dr. Mayez Al-Mouhamed, for his constant help, guidance and the attention that he devoted throughout the course of this work . He was always kind, understanding and sympathetic to me.

I would like to thank co-chairman Dr. Habib Yousef, whose continuous encouragements can never be forgotten. His valuable suggestions made this work interesting and learning for me.

Thanks are also due to my thesis committee member Dr. Hasan R. Barada for his interest, cooperation, advice and constructive criticism.

I also wish to thank the faculty, and the staff members of the Computer Engineering

Department for their support.

Special acknowledgment is due to my friends Naseer, Nisar, Feroze, Sajjad, Ather, Abdul Raheem, Mahboob, Aleem, Abdul Samad, Younus and to all my friends at KFUPM for their moral supprot, good wishes and the memorable days we shared together.

Finally, I thank my parents, brother and sisters for their love, sacrifices, prayers and understanding. They helped me a lot in achieving my objectives.

Contents

List of Figures

List of Tables

THESIS ABSTRACT

Name: MOHAMMED KALEEMUDDIN

Title: EVALUATION OF PIPELINED SWITCH ARCHITECTURES FOR ATM NETWORKS

Degree: MASTER OF SCIENCE

Major Field: COMPUTER ENGINEERING

Date of Degree: June 1997

The rapid evolution in the field of telecommunications has led to the emergence of BISDN (Broadband Integrated Services Digital Networks), to support a variety of communication services. The proposed transfer technique for BISDN to make an efficient use of the communication resources is Asynchronous Transfer Mode (ATM) which is a high-speed packet switching technique. It is the most appropriate technique to use for data and non-data applications with bursty traffic. It offers greater flexibility than circuit switching in handling the wide diversity of data rates and latency requirements resulting from the integration of services. The challenge therefore, is to design and build switches capable of switching relatively small packets at extremely high rates.

In this thesis we study various Banyan-based switch architectures. A performance analysis of pipelined banyan proposed by P.C. Wong [25] is discussed. On the basis of this paper we present the design and analysis of improved pipelined banyans based on dilated banyan and expanded banyan. We have evaluated the performance of these two switches through both analytical model and simulations under Uniform traffic and ATM Traffic.

King Fahd University of Petroleum and Minerals, Dhahran.
June 1997

موجز الرسالة

اسم الطالـــب : محمد كليم الدين

عنوان الرسالة : تقييم تصاميم مفاتيح شبكات ATM المتوالية .

الدرجـــــة : درجة الماجستير

التخصـــص : هندسة الحاسب الآلي

تاريخ الشهادة : يونيه ١٩٩٧م .

لقد أدى التطور السريع في مجال الاتصالات اللاسلكية الى ظهور الشبكات الرقمية المتكاملة ذات الخدمة المطوّرة (BISDN) لدعم خدمات الاتصالات . وقد جاء نظام نقل المعلومات ATM ليكون الدعامة الأساسية لتلك الشبكات في تقنية نقل المعلومات لقدرته على استخدام جميع العمليات المتوفره في شبكات (BISDN) .

هذا النظام (ATM) يعتبر أفضل تقنية متوفرة للاستخدام في نقل جميع تطبيقات نقل المعلومات الناتجة من مصادر ذات تخلخلات عالية . إنه يقدم مرونة كبرى بالمقارنة مع طريقة مفاتيح الدائرة في تنفيذ مختلف السرعات لنقل المعلومات وكذلك في الوقت الكافي لنقل تلك المعلومات الناتج من اتحاد جميع خدمات الاتصالات . يكمن التحدي هنا في تصميم وبناء مفاتيح لها القدرة على توجيه المعلومات الدقيقة بسرعات هائله جدا . درسنا في هذه الرسالة جميع التصاميم للمفاتيح المبنيّة على منتاح (Banyan) . لقد قدم P.C.Wong دراسة تحليلية لأداء مفاتيح Banyan المتوالية [25] ، وعلى أساس تلك الورقة طرحنا التصاميم والتحاليل لمفاتيح Banyan المتوالية والمعدّلة التي تعتمد على التصاميم المكثّرة والموسعة لمفاتيح Banyan . وقد تم تقييم أداء هذه التصاميم عن طريق التحليل والمحاكاة على أساس مصادر المعلومات الموحدة وكذلك مصادر معلومات خدمات ATM .

درجة الماجستير في العلوم

جامعة الملك فهد للبترول والمعادن

الظهران ، المملكة العربية السعودية

يونيه ١٩٩٧م

Chapter 1

Introduction

The two fields of communication, telecommunication and data communication have come close together with the advancements in communication technologies. Originally voice was carried as analog information. With the advent of digital transmission media for voice networks, voice is now often digitized and carried as packets. User demands for the combined voice and non-voice services has led to the development of a new system called Integrated Services Digital Network (ISDN). The primary goal of ISDN was to integrate voice and non-voice services. Initially the services provided by ISDN were limited to narrowband and it was called Narrowband ISDN (N-ISDN). Basically ISDN was an attempt to replace the analog telephone system with a digital one suitable for both voice and non-voice traffic. Some of the features of N-ISDN are, call forwarding, conference calls, display of caller's number, name and address. Soon the services provided by N-ISDN were found to be limited and N-ISDN was extended to Broadband ISDN (B-ISDN). B-ISDN was developed with a target to support high speed services and a complete range of video services in addition to the services of N-ISDN.

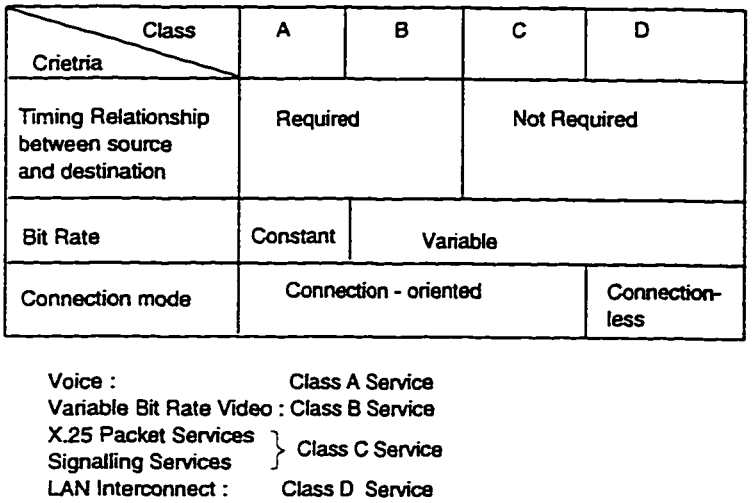

Figure 1.1: Service classes in B-ISDN.

B-ISDN is intended to support a wide range of voice, text, image, data and video services [22],[12]. Some examples of these broadband services are broadband video telephony, video conference, video surveillance, video/audio services, distributed services like EDTV (Extended Definition TV), HDTV (high-definition TV), high-quality TV, pay TV(pay-per view, and pay-per-channel), full-channel broadcast videography etc. [14]. To simplify these wide range of services ITU has classified these services into different types as shown in Figure 1.1 [10].

Synchronous transfer mode (STM) was used to transmit digitized voice. STM technologies used dedicated physical paths that are established when a call is setup [11]. The voice packets are of fixed length and transmitted as a frame over a time division multiplexed (TDM) channel. Each slot is synchronized to the frame bit. Voice transmission needed this synchronization technique because voice traffic would suffer if words arrived in bunches with irregular time spacing. Each time slot represents one voice call and data is identified by its position within a frame. Therefore even if a speaker is silent his or her time slot cannot be grabbed by another user.

Since data does not require periodic transfers like voice, a data circuit could utilize unused time slots whenever they appeared. A more efficient transfer mode for data communication traffic is *packet switching*. Data packets are identified by the address contained within each packet instead of by its position. Packet switching has the characteristic of variable length delay and throughput. An attempt to mix voice and data packets may lead to loss of voice information because voice cannot tolerate high delays and larger data packets may cause more delay to the subsequent voice packets. Therefore pure packet switching or pure circuit switching is not suitable for ISDN which has an objective of integrating services with diverse performance requirements.

Initial ISDN technology called narrowband ISDN is basically digital circuit switching. It could not meet the present demand of a diverse range of services with huge bandwidth requirements. Therefore broadband ISDN has been proposed under the support of a new transfer technique called *Asynchronous Transfer Mode (ATM)*. ATM is a fixed length packet switching technology. The 53-byte length of the ATM (Asynchronous Transfer Mode) cell is a compromise designed to make ATM useful for data as well as voice, video and other real time traffic that cannot tolerate randomly varying transmission intervals and delays. Slicing a long data frame into 48-byte pieces and adding a 5-byte header may introduce significant amount of overhead, but it enables the same kind of technology for any kind of service. Therefore ATM has become the basic transfer mode for B-ISDN. Other benefits of ATM are bandwidth efficiency, multi service capabilities, scalability, flexibility etc. Also an ATM users may send bursts of as many or as few cells as necessary to transfer data and pay only for the cells they send, not for the speed of a dedicated facility, which

they may be using only for a fraction of time.

1.1 Asynchronous Transfer Mode

Asynchronous transfer mode is a high-speed, connection-oriented packet switching and multiplexing technology that uses 53-byte cells, to transmit different types of traffic simultaneously, including voice, video, and data [20]. The basic idea behind ATM is to transmit all information in small, fixed-length packets called cells. The cells are 53-byte long. The advantages of using cell switching over circuit switching are :

1. Cell switching is highly flexible and can handle both constant rate traffic (audio, video) and variable bit rate (data) easily.

2. At very high speeds envisioned (gigabytes per second) digital switching of cells is easier than traditional multiplexing techniques especially using fiber optics.

3. For television distribution, broadcasting is essential. Cell switching can provide this while it is not possible by circuit switching.

In the following we present a review of ATM cell structure and its protocol reference model.

1.1.1 ATM Cell Structure

In ATM all information(data or control) to be transferred is packed in fixed-size 53 octet slots called cells as shown in Figure 1.2(a). These cells have 48 octets of information field and 5 octets of header field for carrying information pertaining to

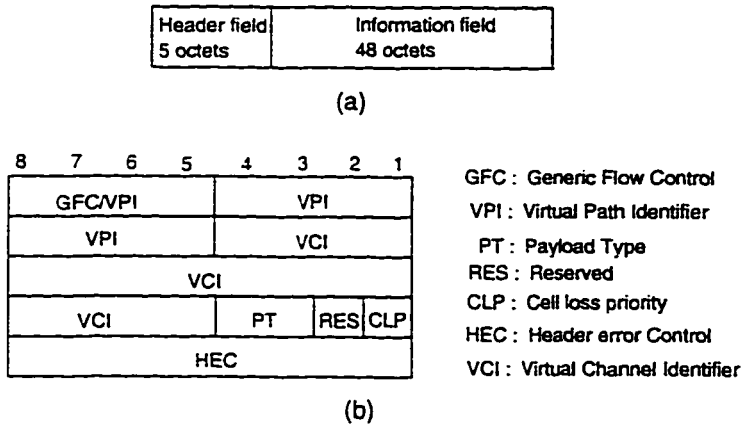

Figure 1.2: ATM cell structure and protocol architecture.

the ATM layer functionality. Header includes information that identifies the virtual channel to which the packet belongs. Octets are sent in increasing order starting with octet 1 (i.e. header) followed by the information field. Bits within an octet are sent in decreasing order starting with bit 8. The structure of header in ATM is shown in Figure 1.2(b). Field GFC (generic flow control) defines flow control in user-network interface (UNI) and the field VPI represents Virtual path in network-network interface (NNI) which is an interface between network nodes. Field PT (payload Type) identifies user information cells and the network information cells to carry information needed by the network for its maintenance and operation. CLP (cell loss priority) is used to explicitly indicate the cell loss priority. Header error control (HEC) contains cyclic redundancy check sequence which is processed by the physical layer to perform error detection and single bit error correction. Cell routing is based on two level addressing structure: the virtual path and virtual circuit identifiers (VPI and VCI). The VPI identifies the physical path, which is then used by a set of VCI's.

1.1.2 ATM Protocol Architecture

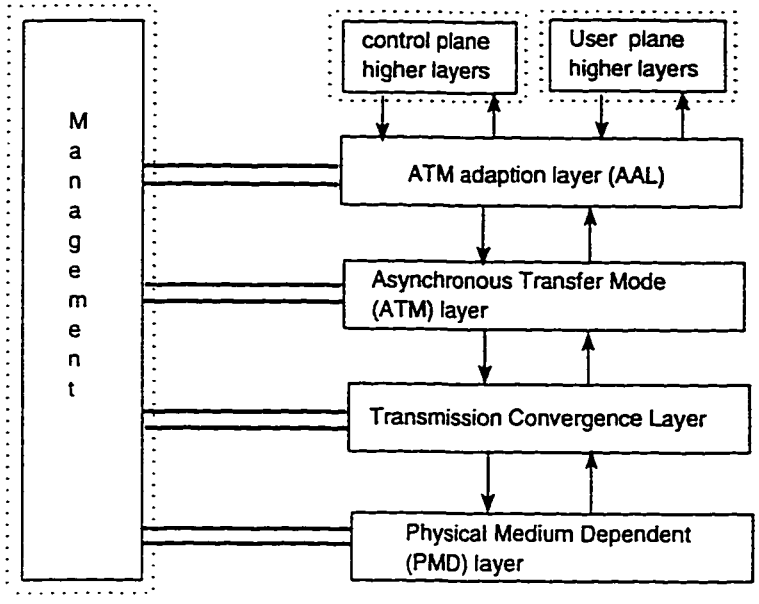

Figure 1.3: Protocol reference model for B-ISDN.

The protocol stack of B-ISDN using ATM is different from OSI model and TCP/IP model. Figure 1.3 shows the protocol model for ATM. As seen it can be described in three logical planes, user plane, control plane and management plane. The user plane coordinates the interface between user protocols, such as IP or SMDS (Switched Multimegabit Data Service) and ATM. The control plane coordinates signalling and setting up and tearing down virtual circuits. The management plane coordinates the layers of ATM protocol stack. These layers of management plane shown in the figure are described in detail below:

Physical Medium Dependent (PMD) layer: This layer defines electrical properties of the carrier signals(such as voltage and wavelength) and physical properties of the media (such as fibre modes and connector structure).

Transmission Convergence(TC) layer: This layer is a provision for adopting

other physical layers for transmitting ATM cells. This layer performs generation and verification of header, frame and cell delineation, and line coding. Cell delineation is determining the cell boundaries for the stream received from physical medium layer. The TC layer receives cells from ATM layer and packs them into appropriate physical medium format. The receiver is always in one of the three states: *hunt*, *presync*, and *sync*. In hunt state, the incoming bit stream is monitored to detect a 5-byte word with a correct CRC. When a match occurs the receiver moves to presync state and searches for δ consecutive matches. If found it moves to sync state, else it moves back to hunt state .

ATM layer: This layer of ATM protocol stack handles most of the processing and routing activities. These include building the ATM header, cell multiplexing/demultiplexing, cell routing using VPIs/VCIs, cell reception and header validation, payload-type identification, quality of service specification, and flow control and prioritization.

ATM Adaptation Layer(AAL): ATM layer does not provide a number of services such as detection of lost cells, means to determine and handle cell delay variation, detection of misinserted cells, information on the frequency of the service clock or content of user information. The reason is that all these services are not required by every B-ISDN application. For example data traffic does not require any information on the frequency of the service clock and voice may not require any awareness on contents of the information. AAL provides these optional services with end-to-end significance. It supports higher layer functions of the user plane and control plane and provides the connection between ATM and non-ATM interfaces. There are two categories of AAL functions, (i) CBR (constant bit rate)

oriented functions and (ii) bursty data services. Some of the CBR functions are cell assembly/dissambly, variable delay compensation, lost cell handling, etc. Some bursty data services include segmentation of information units into cells, handling partially filled cells, and actions on lost cells. Each AAL Protocol has two sublayers: segmentation and reassembly (SAR) sublayer and convergence sublayer (CS). SAR breaks the data into 48-byte cells and collects them into packets. The CS prepares the user data for the lower sublayer.

1.2 ATM Networks

Connection oriented services dedicate a physical path circuit switched to a voice call for the duration of the call and no other call may use the facilities for this time. After the call is complete, everything is torn down and made available for use by the next call. *Connectionless* services at the other extreme end have no path associated with a communication. Each packet in a connectionless service may follow a different path from source to destination, depending upon which links are available at any given instant of time. This makes maximum use of network resources, but adds uncertainty to the time required for information to traverse the network. While this presents no problem to digital data, it could be a problem with real-time services, like voice and video which cannot afford the resequencing delay time. [18].

ATM avoids either extreme. It uses the concept of *virtual networking*. A virtual connection is established between each pair of ATM switches which are needed to connect a particular source to destination by making use of the VPI and VCI fields in the header. This two part addressing allows the network to use a short-hand notation for major trunks between locations while maintaining the identity of

individual circuits within the trunk. A virtual path may consist of several virtual channels. The VPI may identify a trunk between two cities and the VCIs may represent individual calls. Switching elements along the way route all the cells on the basis of first byte of the address (VPI field) and the routing table held in each switch. When a virtual path is established each switch is provided with a set of lookup tables that identify an incoming cell by header address, route it through the switch to the proper output port, and overwrite the incoming address with a new table entry that the next switch along the route will use it as an entry in its routing table. The message is thus passed from switch to switch over a prescribed route, but the route is "virtual" since the facility carrying the message is dedicated to it only while the cell traverses it. Two cells that are ultimately headed for different destinations may be carried one after the other over the same physical path for the common portion of their journey. This virtual nature of ATM provides greater network efficiency.

LAN and WAN over ATM: Traditional LANs such as Ethernet, Token Ring, and Fiber Distributed Data interface (FDDI) share a physical medium. Access to this shared medium is dictated by the rules established as part of the LAN standards. The benefits of shared media are low costs, simple physical wiring, and ease when making changes. Drawbacks of shared-media include one-at-a-time access to the medium, the fact that all stations run at the same rate regardless of the need and loss of throughput during heavy use.

An ATM LAN replaces the shared medium with a centralized switch that has a dedicated connection to each user. Control of the network resides in the switch, which routes messages and controls access in the event of congestion. Since each

port is dedicated to one user, the users do not have to contend for access as with conventional LANs. The user has full access to the connection and can send his data in one full burst instead of segmenting the data into shorter lengths as required in other LANs. ATM allows different users to communicate at different rates over different media. For example a user might use 155 Mb/s optical-fiber channels to the switch, while others might use lower-cost 1,544 or 51 Mb/s copper. This versatility makes it easier to tailor the network to specific needs without building in lots of excess capacity. The architectural issues of emulating LAN over ATM is described in [32], [27]. It is easy to provide wide-area network(WAN) to a LAN through another port on the LAN switch. Conventional LANs depend on separate tools such as a bridge or a router to convert LAN rates and protocols to WAN compatible formats. In [9] connectionless service over ATM is proposed to interconnect connectionless LAN'S and MAN's.

1.3 Conclusion

In this chapter we have discussed the basic concepts of ATM and its benefits. To summarize ATM handles a mix of delay-insensitive, loss-insensitive, delay-sensitive, and/or loss-sensitive traffic over the same ATM interface and network infratucture. It combines the high speeds of circuit switching over a single simplified network infratucture. ATM is highly flexible and scalable - allowing support ranging from small private networks to very large public networks [21]. It is the most appropriate transfer technique for B-ISDN. In the next chapter we will discuss the ATM with specific to switching architectures.

Chapter 2

Literature Survey

2.1 ATM Switches

Switching refers to means by which the limited transmission facilities are allocated to the users so as to provide a certain degree of connectivity among them. An ideal packet switch is a box that can route all packets from their input lines to their requested output lines without loss and with minimum transit delay while preserving the order in which they have arrived to the switch. In addition to this, sometimes switches may be required to perform two more functions [30]. The first is a *multicast* function operation. Depending on the application being serviced, it may be necessary for a set originating at a source node in the network to be routed to more than one destination. There are several ways of achieving this. One is to create multiple copies of this cell at source and independently route each of them to desired destinations. Another way is to replicate the cell at several output ports. The second function is *priority function*. This function may be used to distinguish packets according to priority information provided in them, and to give preferential

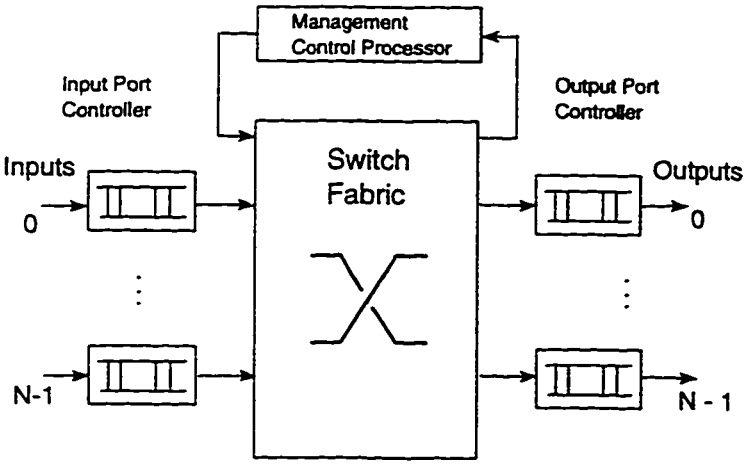

Figure 2.1: Abstract model of an ATM switch.

treatment to higher priority packets.

2.1.1 Abstract Model of an ATM Switch

An ATM switch consists of a set of N input and N output ports, a switch fabric, and management and control processor (MCP) as shown in Figure 2.1 [26].

Input/output controllers: Each port is managed by an intelligent controller. Input port controllers typically provide buffering, cell duplication for multicasting, cell processing, VCI translation, multiplexing traffic from several low-speed devices and path connection requests and reservation through the switch fabric. Similarly the output controllers provide buffering and VCI translation, as well as demultiplexing, and an N:R selection (selecting R packets from a maximum of N for buffering).

Switch fabric: This is a mechanism that routes cells from input to output ports.The functions of the switch fabric is to establish a path between input and output ports within the switch, service discipline for input ports, provide contention resolution scheme(s) to deal with internal blocking, and support several input-output

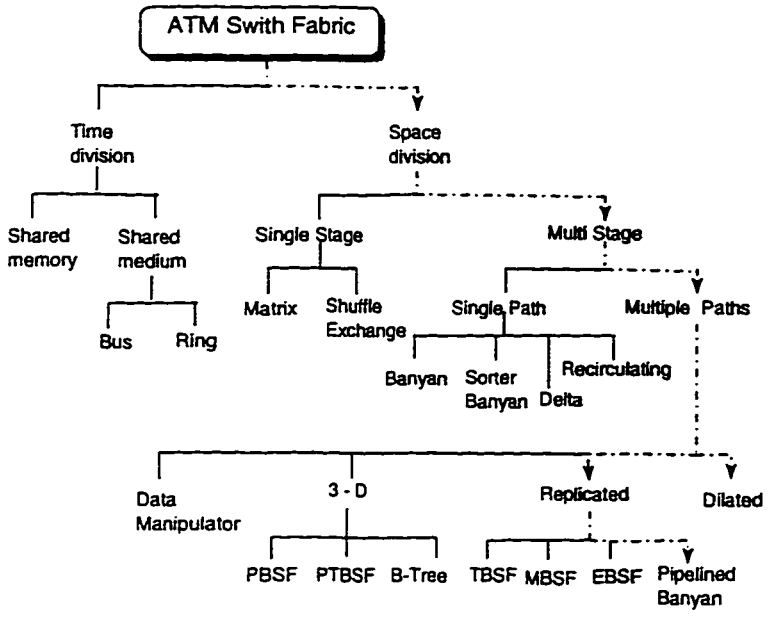

Figure 2.2: Classification of the ATM switch fabric.

port connections.

Management Control Processor: The MCP's function is to communicate with port controllers and facilitate switch operation, administration, and management.

2.1.2 Classification of Switching Architectures

Figure 2.2 shows a classification of Switch architectures [34]. Basically the switching architectures can be classified into two classes, *Time-division switch* architectures and *Space-division switch* architectures. In *time division switches*, the physical resource (such as conducting medium or memory) is multiplexed among several input-output connections, based on discrete time slots. For example buses and memories can accommodate time-division multiplexing. In *space-division switches*, there is no sharing of resources among multiple paths, i.e. multiple concurrent paths are established from input to output port. In time-division switches, control of the

switch is centralized whereas control may be distributed throughout the switching fabric in space-division architectures. Time-division and space-division multiplexing can be combined. For example several time-division switches (clusters) may be interconnected via a space-division switch in a hierarchical fashion.

2.1.3 Time Division Switch Architectures

Shared Memory Switches

The switch consists of a single dual ported memory shared by all input and output lines. Packets arriving on all input lines are multiplexed into a single stream which is fed to the common memory for storage. Internal to the memory, packets are organized into separate output queues one for each output port. Simultaneously an output stream of packets is formed by sequentially retrieving packets from the output queues, one-per-queue. The output stream is then demultiplexed and packets are transmitted on the output lines. There is a central controller which sequentially processes all N incoming packets, determines where to queue the packets and issues the proper control signals. The major disadvantage of these switches is the requirement of memory bandwidth which should be large enough to accommodate simultaneously all input and output traffic. If N is the number of ports and V is port speed, then the memory bandwidth must be $2NV$ [19],[24]. For example a 32-line switch with line speeds of 150 Mbps the memory bandwidth must be at least 9.6 Gbps. This large memory bandwidth may be achieved by parallel memory organization, however there is a limitation on the number of memory banks that can be used in parallel due to centralized control. Also memory access time may become a bottleneck.

Shared Medium Switches

In shared medium switches, all packets arriving on the input ports are synchronously multiplexed onto a common high-speed medium, such as a parallel bus, of bandwidth equal to N times the rate of a single input line. Each output line is connected to the bus via an interface consisting of an address filter and output FIFO buffer. The address filter in the interface on each line reads the virtual address (output address) and determines whether the packet is to be written into its FIFO buffer. There is a single path through which all packets flow and it operates as a broadcast time-division bus. Demultiplexing is done by the address filters in the output interfaces.

2.1.4 Space Division Switch Architectures

Fully connected switches:-

Crossbar switch is a single-stage single-path non-blocking (internal conflicts only) switching fabric, originally developed for circuit switching. As shown in Figure 2.3(a) an $N \times N$crossbar switch has an array of N^2 individually operated crosspoints, one for each input-output pair. These crosspoints could be electromechanical relays or semiconductor switches. Each crosspoint has two possible states: cross(default) and bar. A connection between input port i and output port j is established by setting the (i,j)th crosspoint switch to the bar state. The advantages of crossbar switches are, non-blocking with respect to internal conflicts, architectural simplicity, and modularity. The drawbacks are square growth complexity (i.e. the number of crosspoints grows as $O(N^2)$), different input-output pairs may have different transit delays and the switch suffers from output conflicts.

The **knockout switch** [35],[36] as shown in Figure 2.3(b) broadcasts all incoming

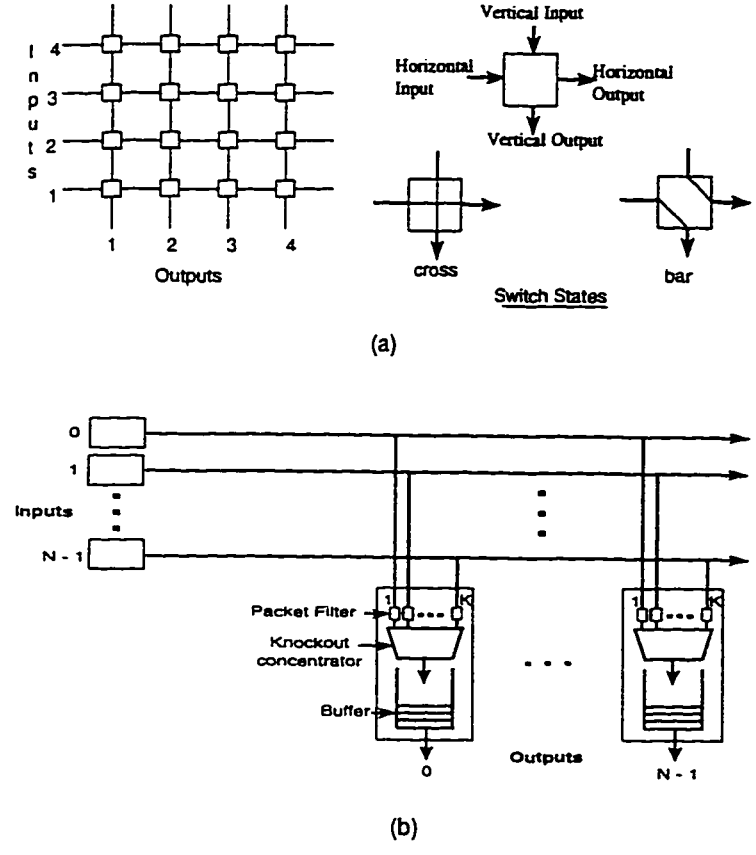

Figure 2.3: Fully-connected switch fabrics. (a)Crossbar switch, (b)Knockout switch

cells to outputs. Each output port has a *bus interface* that performs several functions. It filters cells not intended to that particular port. It has a concentrator which selects only L out of N cells which may be destined to that port in a given time slot. Finally it performs buffering using a FIFO buffer that is shared among all L lines in a clever arrangement. A *cell loss probability* smaller than 10^{-6} at a load 0.90 can be achieved with $L = 8$ for an arbitrary large value of N. But the delay becomes very large when the load exceeds 0.9. Despite its excellent performance, the knockout switch is complex for large switch sizes. It requires N^2 distinct physical paths, N^2 address filters, and $N(N \times L)$ Knockout concentrators.

Banyan Based Switches:-

This class of switches is based on *multistage interconnection networks (MINS)* [3]. In multistage switches, the switching elements are arranged in multiple stages. A N-input-port switch composed of $b \times b$ simpler switching elements, has K stages and N/b switches in each stage where $N = b^K$. Hence complexity of this class of networks is of the order of $Nlog_bN$. These architectures are suitable for VLSI implementation due to their modular design. A *banyan network* is attractive for its simple routing scheme, low hardware complexity, regular structure and ability to deliver multiple packets simultaneously [1].

Simple Banyan Switch: A banyan network consists of $k = log_2N$ stages each comprising $N/2$ binary switching elements which can assume two states cross and straight as shown in Figure 2.4(b). The interconnection lines between the stages are placed in such a way as to follow a unique path from each input to each output. There are several isomorphic configurations [4],[5]that such interconnection networks may take, some of which are defined in Table 2.1. The permutation for each of these networks are shown in Table 2.2.

An example of Banyan switch based on the Omega network is shown in Figure 2.4(a). This has 8 input and output ports and three stages. There is always a unique path for each pair of input-output. Since there is only one path for each input-output pair, banyan network switches are classified as single path switches. The disadvantage of these switches are *internal blocking* and *external blocking*. Internal blocking occurs when two cells are contending for the same output link resulting in internal conflict at any stage. External blocking may occur when more than one packet arriving in the same time slot are destined to the same output port. The following switches

Network	Recursive definition	pos_i
Omega	$\Omega_n = (\sigma E)^n$	$s_{n-i-1} \ldots s_0 d_{n-1} \ldots d_{n-i}$
Indirect Omega	$I\Omega_n = (E\sigma^{-1})^n$	$d_{i-2} \ldots d_0 s_{n-1} \ldots s_i d_{i-1}$
Baseline	$B_n = (\Pi_{j=n}^{j=2} E\sigma_j^{-1})E$	$d_{n-1} \ldots d_{n-i+1} s_{n-1} \ldots s_i d_{n-i}$
Indirect Baseline	$IB_n = E(\Pi_{j=2}^{j=n} \sigma_j E)$	$s_{n-1} \ldots s_i d_{n-1} \ldots d_{n-i}$
Cube	$C_n = \sigma E(\Pi_{j=n}^{j=2} \beta_j E)$	$d_{n-1} \ldots d_{n-i+1} s_{n-i-1} \ldots s_0 d_{n-i}$
Indirect Cube	$IC_n = (\Pi_{j=2}^{j=n} E\beta_j)E\sigma^{-1}$	$s_{n-1} \ldots s_i d_{i-2} \ldots d_0 d_{i-1}$
Delta	$\Delta_n = (E\sigma)^{n-1}E$	$s_{n-i} \ldots s_1 d_{n-1} \ldots d_{n-i}$
Indirect Delta	$I\Delta_n = (E\sigma^{-1})^{n-1}E$	$d_{i-1} \ldots d_1 s_{n-1} \ldots s_i d_i$

Table 2.1: Some multistage networks.

Perfect shuffle	$\sigma(x_{n-1}, \ldots, x_0) = x_{n-2}, \ldots, x_0, x_{n-1}$
Sub-shuffle	$\sigma_{(i)}(x_{n-1}, \ldots, x_0) = x_{n-1}, \ldots, x_i, x_{i-2}, \ldots, x_0, x_{i-1}$
Bit reversal	$\rho(x_{n-1}, \ldots, x_0) = x_0, x_1, \ldots, x_{n-2}, x_{n-1}$
Butterfly	$\beta(x_{n-1}, \ldots, x_0) = x_0, x_{n-2}, x_{n-3}, \ldots, x_2, x_1, x_{n-1}$
Sub-butterfly	$\beta_{(i)}(x_{n-1}, \ldots, x_0) = x_{n-1}, \ldots, x_i, x_0, x_{i-2}, \ldots, x_1, x_{i-1}$
Exchange	$e(x_{n-1}, \ldots, x_0) = x_{n-1}, \ldots, x_1, \bar{x}_0$
Sub-exchange	$e_i(x_{n-1}, \ldots, x_0) = x_{n-1}, \ldots, x_i, \bar{x}_{i-1}, x_{i-2}, \ldots, x_0$

Table 2.2: Some known permutations.

show some techniques adopted to solve these internal/external blocking problems.

Sorter-banyan switches (Figure 2.5(a)): These type of switches use the technique called *load-distribution* to solve the problem of internal blocking in banyan networks. A sorter network is used to sort the input traffic based on the destination ports and present them to the Banyan switch on adjacent input ports. Switches which use Batcher-Bitonic sorter as a sorter network are called Batcher-Banyan switches [17],[23] . Although the technique of *load-distribution* resolves the internal conflict, blocking at output ports is still possible.

Starlite architecture [2]: This switch uses another technique called *recirculation*, in addition to load-distribution as described above. As shown in Figure 2.5(b).

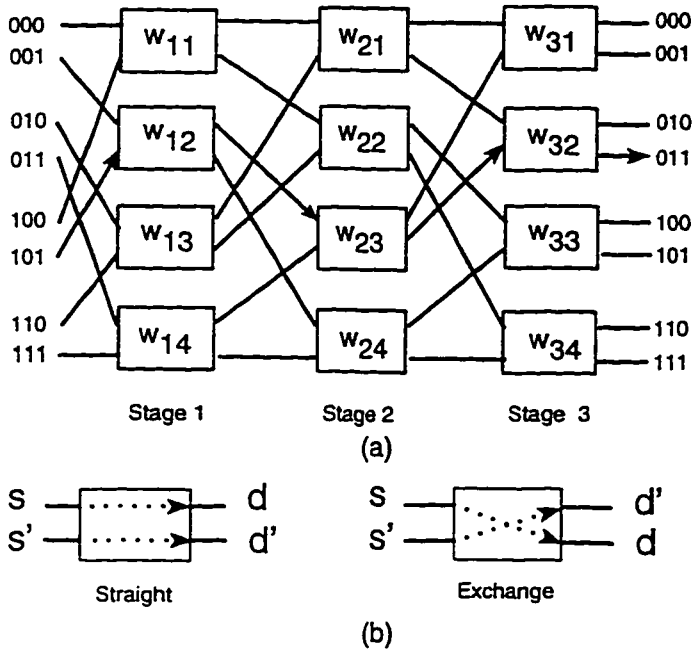

Figure 2.4: Banyan switch based on omega network. (a) 3-stage omega network, (b) states of a 2 × 2 switch

a trap network examines the output of the sort network and removes all packets (except one) with same destination address. The duplicate-destination packets are routed back to the sort network to be issued in the next group selection and sorting cycle. The routed back packets are placed on the empty input ports which subject the switch to out-of-sequence delivery.

Sunshine Switch [13]: This switch employs several of the above mentioned techniques in order to increase the throughput. The architecture (shown in Figure 2.5(c)) consists of a Batcher sorting network at the input. The output of the Batcher network consists of cells sorted on the basis of their destination address. The trap network which follows, selects at most K cells per destination address to be routed. The remaining cells are separated by the concentrator and forwarded to the recirculation buffer. The recirculation buffer delivers these cells to dedicated

input ports for transmission in the next cycle. The recirculation buffer consists of T parallel paths to the input of the Batcher network with one unit of delay. At the output there are K parallel Banyan networks. The selector delivers the cells to the Banyans to be routed. Each input and output port is managed by a controller IPC and OPC respectively.

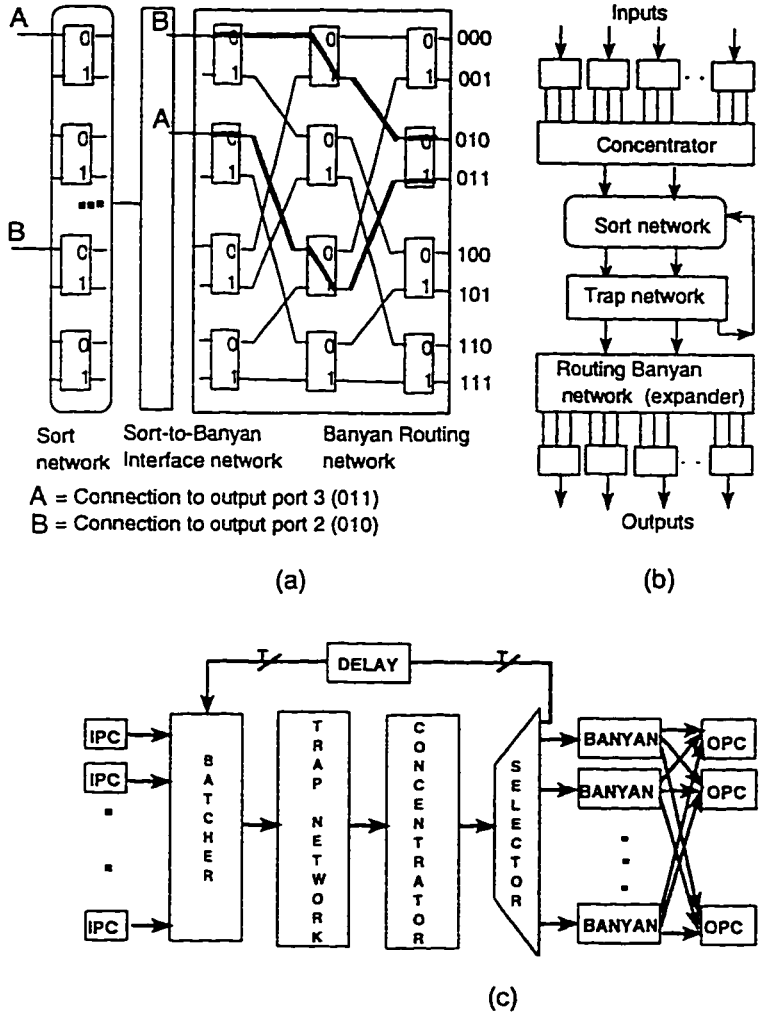

Figure 2.5: Modified banyan switches. (a) sorter-banyan switch, (b) starlite switch, (c) Sunshine switch

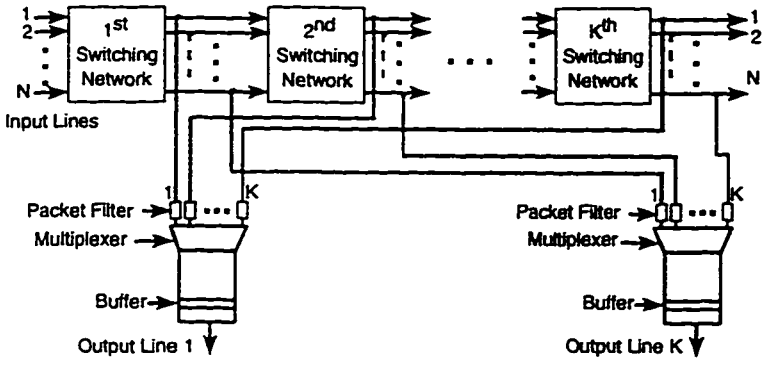

Figure 2.6: Tandem banyan switch fabric.

Multiple Banyan Switch Fabrics:-

The switches described in previous section used only one banyan network. Here we will discuss switches which use multiple copies of the same banyan network in various configurations.

Tandem banyan Switch Fabric (TBSF): Tandem Banyan [31] tries to overcome the internal/output conflicts in simple banyan networks by placing multiple copies of banyan network (say K) in series as shown in Figure 2.6. Output of every banyan network is connected to both corresponding inputs of the following network in series and the corresponding output buffer, with the exception of last banyan for which outputs are only connected to the output buffers. Whenever there is a conflict between two packets at some switching element, one of the packets is routed properly and the other is marked as misrouted. Packets marked as misrouted are given least priority and thus will not affect the routing of properly routed packets at later stages of that network. The properly routed packets are extracted from the fabric and placed in corresponding output port buffers, while the misrouted packets are fed into the next banyan. This process is repeated through the K banyan net-

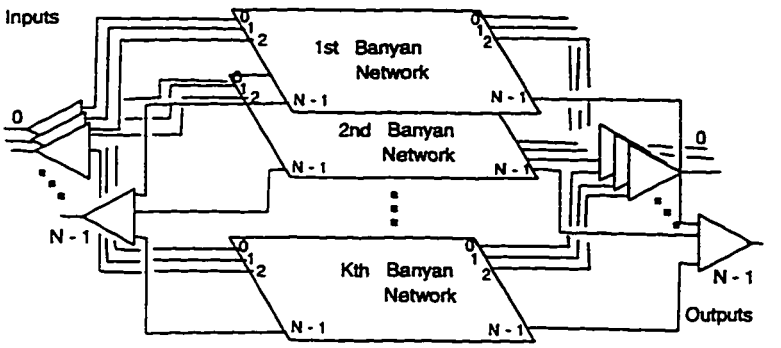

Figure 2.7: Multi banyan switch fabric.

works. Unsuccessful packets at the last banyan are lost. We note that the load on successive banyan networks decreases and so does the likelihood of conflicts. With a sufficiently large K, it is possible to decrease the packet loss to the desired level. The drawback of this approach is the relatively large switching delay caused by the longest path through the K-banyan. During this time no cells, among those which have been correctly routed earlier, can be processed to avoid out-of-sequence delivery.

Multi Banyan Switch Fabric (MBSF): This switch comes under the category of parallel switch architectures. It has multiple planes of banyan networks arranged in parallel planes as shown in Figure 2.7. The inputs and output ports are connected to these multiple instances (planes) of the fabric. Each input port can distribute its traffic to multiple homogeneous fabric planes. Similarly each output port can be fed from multiple fabric planes. The input traffic is randomly distributed on these multiple parallel banyan networks. Though the switching time is relatively small, the cell-loss ratio saturates at some level (0.9) that is not acceptable for some ATM traffic types.

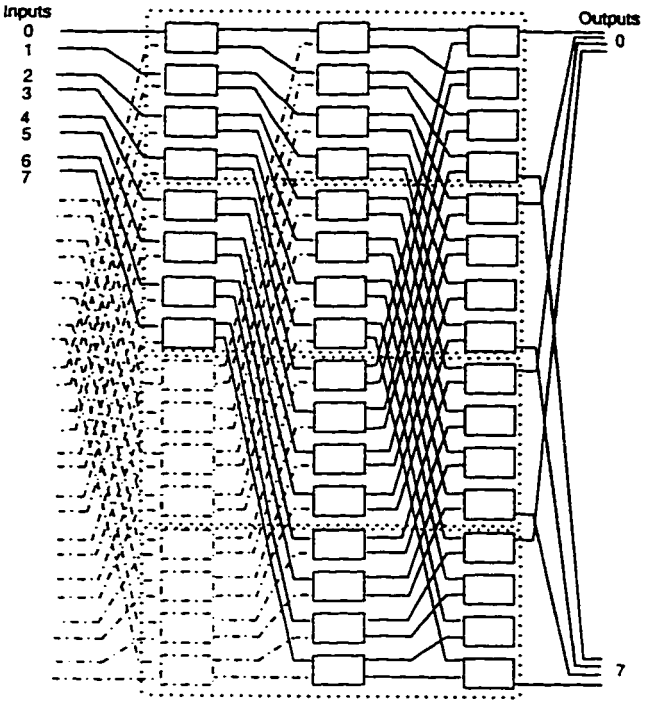

Figure 2.8: 8 × 8 Expanded banyan switch fabric with expansion factor 4.

Expanded Banyan Switch Fabric (EBSF): Expanded banyan switch fabric expands the output capacity of each port by EF times where EF is referred to as expansion factor. An EBSF of size $N \times N$ is constructuted by interleaving EF $(N \times N)$ banyan networks. If we let $M = N \times EF$ then the switch has $M/2$ switching elements per stage and $log_2 N$ stages. An example of 8 × 8 EBSF with expansion factor 4 in shown in Figure 2.8. As the value of EF increases more and more switching elements and links of the first stages become idle. and can be removed without effecting the proper functioning of the switch. If we assume that the expansion factor is a power of 2 i.e. $EF = 2^k$ then a conflict path is guaranteed in the switch for k stages. In was shown in [7] that high throughputs can be achieved by increasing the values of EF by a reasonably small value.

3-D Banyan Switch Fabrics:-

In this section we will discuss some switch architectures based again on parallel banyan network but with three dimensional structure. The cells in these-architectures have the potential of horizontal routing across planes as well as vertical routing from one plane to the other.

Piled Banyan Switch: Piled banyan switch fabric (PBSF)[15] consists of banyan networks connected in a three-dimensional pattern. Each switching element, except in the highest and lowest layers, provides four inputs/outputs, two for horizontal and two for vertical direction. In PBSF packets are inserted into the inputs of the highest layer banyan network and routed in the horizontal direction. When two packets collide one of the packets is routed correctly (horizontally) and the other packet is fed to the corresponding switching element in the next lower layer banyan network with a clock delay. When three packets collide (one from the vertical and two from the horizontal direction) one of the packets is routed correctly, the second is routed vertically down and third packet is treated as "dead-packet". Since the conflicting packets are routed to the same position of the next layer down, the effect of routing the packets until this stage is not lost in PBSF. This is the reason that pass-through ratio of PBSF is larger than TBSF where a conflicting packet must be routed from the first stage again in the next banyan. However in the case of three packets colliding at a switching element in PBSF one of the packet is treated as dead-packet and never saved. This results in saturation of pass-through ratio to 98% in PBSF even if large number of banyans are used. Whereas if sufficiently large number of banyans are used in TBSF a pass-through ratio greater than 99.9% can be achieved.

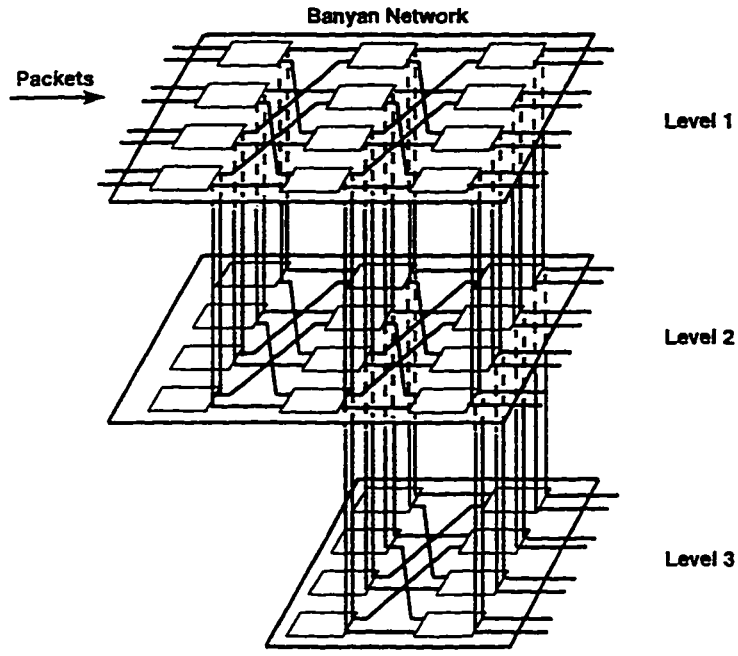

Figure 2.9: Piled banyan switch fabric.

B-Tree Switch: This is a self-routing fault-tolerant switching network for ATM. It is called B-Tree as it is implemented by coupling multiple *baseline networks* and because it has a binary tree structure [16]. Figure 2.10 shows the basic structure of $N \times N$ B-Tree network. It has N input ports and N output ports and $n = log_2(N)$ Number of stages. Each cube $[i, j]$ in the figure represents a stage in column-i and row-j for all $0 \leq i, j \leq n - 1$. Each stage is implemented by $N/2 = 2^n - 1$ *Switching Elements*(SE). Each 4×4 SE has 4-inputs I_0, I_1, I_2, I_3, 2 - formal outputs f_0, f_1 and two redundant outputs r_0 and r_1. No buffer is employed in the SE. Each parallelogram between two sages describe interconnection between stages. Due to the recursive structure of the baseline network it is very easy to expand a B-tree. To obtain a $2N \times 2N$ B-Tree place two $N \times N$ B-Trees one below the other, re-label each stage $[i, j]$ as $[i + 1, j]$ in these two B-trees, add a $2N \times 2N$

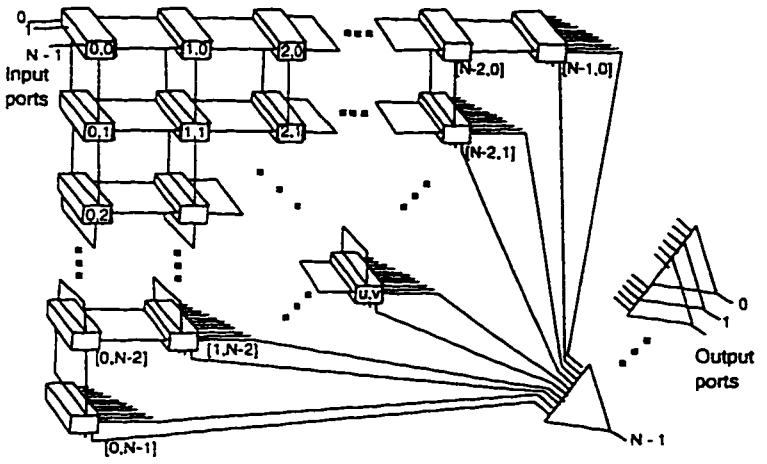

Figure 2.10: B-Tree switch fabric.

baseline network at first stage, mark each of its stage as $[0, j]$ from top to bottom and finally perform shuffle between stages $[0, j]$ and $[1, j]$, for $0 \leq j \leq n-1$.

The simulation results given in the paper [16] show that decrease of performance is very small with increasing size of network. Therefore the network is suitable for large scale switches. When results under fault conditions were observed throughput was found to decrease with the increase in the number of faulty SE's. However this decrease of throughput is significant only in small networks and it is negligible in large networks. The basic advantage of B-tree is that it is *fault-tolerant*. A single fault in any other switch can destroy whole routing function. Whereas a $N \times N$ B-Tree provides $2n$ access points for each output port. Therefore each port has the capability to accept upto $2n$ cells simultaneously if there is no internal conflict. There is also no head-of-line (HOL) blocking due to this property in B-Tree. B-Tree provides N-alternative paths between each input/output pair. This large number of paths contributes to improving reliability. To tolerate faults in first stage a B-Tree(τ) can be used. The switch has ideal performance even in presence of faults in

the switch. Features such as only one type of SE, very little complexity and modular structure of the architecture make the implementation of B-Tree Switch very easy. The drawbacks are performance measurement of the switch was done under uniform traffic and not non-uniform ATM traffic. Also the switch is sensitive to faults at first stage and last stage. For the scalability of the switch we can use only Baseline network and not others such as Omega network etc.

Parallel Tree Banyan Switch Fabric (PTBSF): This switch architecture was proposed recently to overcome some of the problems encountered in TBSF [6]. In TBSF cells misrouted in a banyan are applied at the first stage in the next banyan. Thus the routing effort in the previous banyan is lost. For example cells conflicting in the last stage, are applied again at first stage stage in next banyan and thus lose the routing effort until last stage in the previous banyan. A PTBSF as shown in Figure 2.11 has banyan networks arranged in the form of parallel tree structure. The switching elements used have a 4×3 structure i.e. it has four inputs and three outputs. All cells are applied at the topmost banyan. A conflicting cell is routed down either to the left banyan or to the right banyan. There can be at the most three different cells at the input of any switching element (SE) contending for the same output link. Since there are three output lines one is routed correctly (horizontally), one is routed to the left banyan and other to the right banyan down. Thus internal blocking is completely eliminated and cell loss can occur only at the output ports in the lowest level of banyans. This switch has been shown [6] to perform better than TBSF with respect to switching delay and delay jitter. Generally its performance is better only for small size switches ($\leq 32 \times 32$). Though the switch has proved to be robust under uniform and ATM traffic (cell loss $\leq 10^{-6}$) its hardware requirement

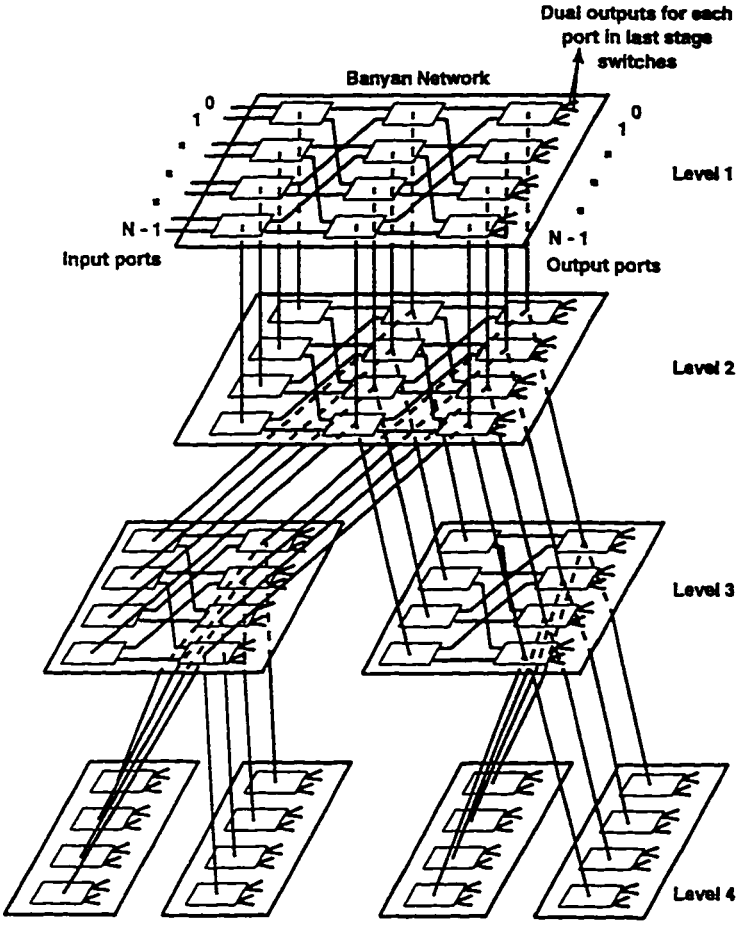

Figure 2.11: Parallel tree banyan switch fabric.

is large inspite of its low delay and delay jitter.

Buffered Banyan Switch Fabrics:-

All the space division switches based on banyan networks discussed until now are non-buffered architectures. In these architectures, cells are applied in one cycle and routed through the switch fabric. Those conflicting cells which are marked "dropped" and which do not reach final output are lost. Therefore basic non-buffered banyan networks are inherently blocking either due to internal or external conflicts

for the cells being routed. In Buffered-Banyan, buffers are added at the inputs of each switching element. When a conflict occurs, one packet is forwarded while the other is kept in the buffer. The throughput can then be increased. To avoid buffer overflow, flow control mechanisms are implemented among stages of switching elements.

Helical Path Switch: Figure 2.12 shows the block diagram of an 8×8 helical switch [33]. It is composed of three major components: the broadcast unit, the FIFO buffer, and the non-blocking concentrator. The cell format as shown contains $log_2 N$-length address bits, a cell identity bit and the information payload. Dummy cells (control packets) are introduced to prevent out-of-sequence delivery. The identity bit is used to identify whether the cell is real (0) or dummy (1). The broadcast unit is used to prevent any cell from overtaking prior cells as they hop from one stage to the next stage. Each broadcast unit is a 1×2 element. It broadcasts any incoming cell on both the outputs and sets the their identity bit depending on the destination routing bit as shown in Figure 2.12. The concentrator transfers the real cells from N' input lines to the $N'/2$ output lines and removes as many dummy cells as possible from its input buffers. In each time slot, the concentrator examines the head-of-line (HOL) cells in its input buffers in a round-robin fashion, transferring real cells to its output, while discarding dummy cells.

Helical switch performs better than buffered banyan switch under both uniform and non-uniform traffic with respect to both delay and throughput. Maximum achievable throughput is around 0.8 under uniform traffic. The main feature of this switch is very low delay is achieved due to switch's buffer sharing capacity. The drawbacks are:

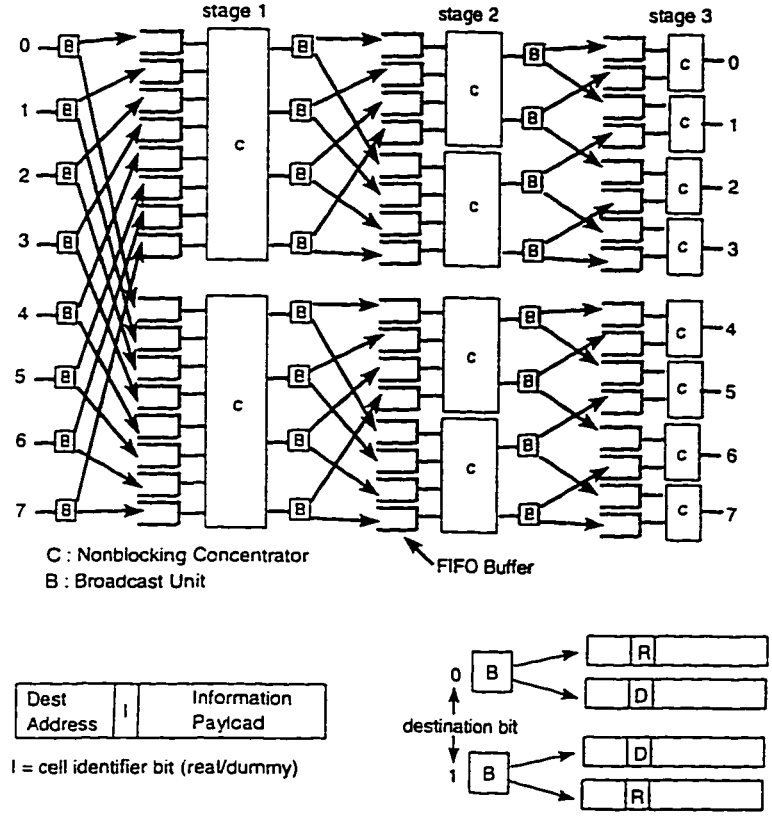

Figure 2.12: Helical path switch fabric.

- The cells are moved sequentially from inputs to outputs of the concentrators
- The switching delay increases with increase in the load which lead to high delay when load approaches 1.

Pipelined Switch: Figure 2.13 shows a Pipelined switch fabric [25] with N input and N output ports. It has one single control plane and a number of data planes all with the same topology. The control plane is for path reservation and the data planes are for actual packet transmission. The time is divided into reservation slots and one data plane is selected in each of these slots. Routing address headers of all the incoming packets are stored into routing controllers (RC) at the control

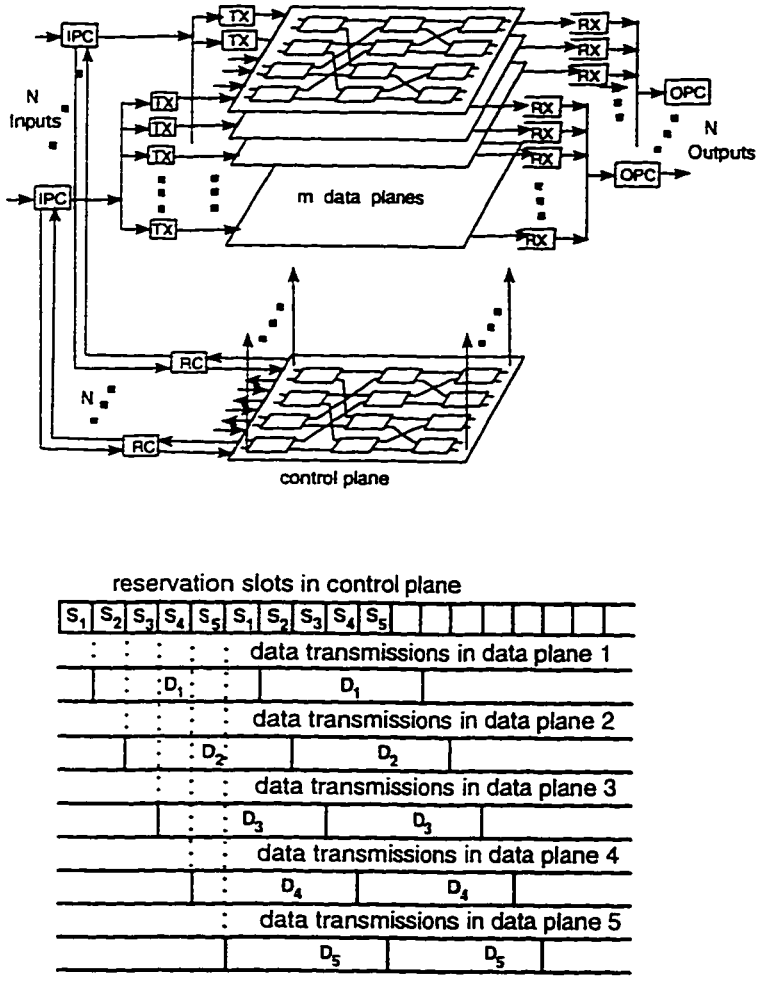

Figure 2.13: Pipelined switch fabric.

plane. These headers of the head of the line (HOL) cells are transmitted into the control plane and self-routed to their destinations. In the case of conflict of two headers, one of them is selected for further routing while the other is dropped. For all those headers which reached their destinations successfully a reverse path is set up and an acknowledgment is sent back to notify which RC has succeeded in making reservations. These RC's in turn remove these successfully routed headers and send an enabling signal to the corresponding IPC. While RC was busy performing the

path reservation the IPC writes a copy of the HOL packet into the transmit register (TX) of the selected data plane. Only those IPC's which receive an enable signal from their RC's transmit the data packets through the data plane. Since the header in ATM cell is much shorter than a packet, multiple reservations can be done during one packet transmission time. Thus the reservation and packet transmissions can be done in a pipelined fashion.

Performance analysis has shown [25] that pipelined banyan performs better than other banyan switches such as TBSF and MBSF. For example to achieve a packet loss probability around $10^{-}6$ in a switch of size 32×32 a pipelined banyan requires 4 banyans while a TBSF requires 6 banyans. It was also shown that the packet loss probability decrease further with increase in the number of data planes or by increasing the input buffer size.

2.2 A summary of previous switch architectures

Basically the switch architectures are classified as time division and space division switches. Time division switches use memory elements to perform routing of cells. The drawback of time division switches is that the size and speed of the switches are constrained by memory bandwidth and access speeds. Therefore space division switches are preferred, which although consume more hardware than time division switches, they achieve better performance with respect to scalability and speed.

Space division switches range from fully-connected switches to switches based on MINs with various configurations. Fully-connected switches provide non-blocking paths, i.e. there is no internal blocking, but the cost of the switch increases quadratically with its size. Examples of fully-connected switches are the crossbar switch and

the knockout switch. Cost of crossbar switch is of order of N^2 which becomes unreasonable when N is large.

To reduce the cost of switch hardware, multistage interconnection networks (MINs) are preferred as building blocks in the design of ATM switches. The advantages of interconnection networks are self routing with distributed control. They have a modular and scalable structure which makes them suitable for VLSI implementation.

Since these switches are not fully-connected, they are blocking unlike the fully-connected crossbar switches. Here more than one packet may contend for the same link inside the switch resulting in an internal blocking, or may contend for the same output port resulting in an external blocking.

Therefore the objectives to be considered while designing a fast packet ATM switch are to minimize blocking and maximize switching speed, throughput and overall performance.

Several techniques were adopted to reduce blocking in the switch. Some of these are discussed below and illustrated with examples.

One of the simplest techniques to reduce blocking is *recirculation* of packets. When more than one cell contend for the same link (internal conflict) or output port (external conflict) one of these conflicting packets is removed from the switch. These removed cells are recirculated and re-inserted again at the inputs of the switch in the next cycle. To reduce the possibility of internal blocking, a distribution (randomization) network may be added in front of the switch. This technique is also called *load distribution*. An example of this is batcher banyan network which consists of a sorting network preceding the routing banyan network. This technique

eliminates the blocking due to internal conflicts but conflicts at the output ports are still possible.

The Starlite switch architecture is an example which uses the above two techniques of load distribution and recirculation to improve throughput. In this case a trap network is inserted between the sorter network and routing banyan. The trap network removes the duplicate destination packets and recirculates them to the sort network to be tried in the next cycle thus trying to solve the output conflict problem. Both of the techniques discussed above reduce the internal and external conflicts. However, they not only increase the hardware and cell delay, but also create out of sequence problems when input cells are not re-issued at the same input port where they were originally.

All the above switches are based on MIN's which are single path switches i.e. there is only one path for each source destination pair. Switches based on MIN's with multiple outlets can support higher bandwidth by passing multiple cells to the same destination. Such switches with multiple paths can be constructed either by adding more stages or by placing banyan networks in series or in parallel. This leads to two types of switch architectures: horizontal MIN switch architectures and vertical MIN switch architectures. The Tandem banyan switching fabric (TBSF) that is shown in Figure 2.6 is an example of a horizontal architecture. TBSF has multiple Banyans connected in series. Each output of every banyan network is connected to the corresponding output port and input of next banyan network in series (except for last banyan). The cells marked as misrouted in a banyan network(due to internal or external conflicts) are fed to the next banyan.

A problem with TBSF is that the switching time of packets is different depending

on the number of banyan networks through which the packet is routed.

For a network of size N the delay in TBSF is as large as $K \times log_2 N$ where K is the number of utilized banyan networks. The cost of hardware also increases as we increase the number of banyans in tandem to improve throughput.

An example of vertical switch architecture is multi banyan switching fabric (MBSF). This consists of multiple planes of banyan networks arranged in parallel. The input and output ports are connected to these multiple instances (planes) of the fabric. Each input port can distribute its traffic to multiple homogeneous fabric planes. Similarly each output port can be fed from multiple fabric planes. The input traffic is randomly distributed on these multiple parallel banyan networks.

The pass through ratio of MBSF is smaller than that of TBSF since multiple networks are used independently in MBSF. More importantly, the throughput of MBSF saturates around 0.9 whereas adding more planes does not lead to noticeable increase in the throughput. This saturation is the main drawback of MBSF.

A non-blocking switch such as crossbar has a maximum passthrough ratio of 0.65, while the TBSF achieves much higher passthrough with only 2 banyans arranged in series. Even with MBSF three banyan networks are enough to overcome non-blocking networks. This demonstrates the advantage of providing multiple outlets and parallel banyan topology.

Another architecture proposed recently is the piled banyan switch fabric (PBSF) that has parallel banyan planes with the possibility of routing cells downward from one plane to the next. based on combination of horizontal and vertical switch architectures. This consists of banyan networks connected in three dimensional topology. Each switching element except in the highest and lowest layers provides

four inputs/outputs, two for horizontal direction and two for vertical direction.

In PBSF packets are inserted into highest layer banyan network and routed in the horizontal direction. When two packets collide, one of the packets is routed correctly and the other packet is fed to the corresponding switching element in the next lower layer banyan network with a clock delay. When three packets collide (one from the vertical and two from the horizontal direction) one of the packets is routed correctly, the second is routed vertically down and the third packet is treated as "dead-packet". Since the conflicting packets are routed to the same position of the next layer down, the effect of routing the packets until this stage is not lost in PBSF. This way PBSF overcomes the problem of large delay jitter in a TBSF where the misrouted cells are routed again from the first stage in the next banyan. However in the case of three packets colliding at a switching element in PBSF one of the packet is treated as dead-packet and never saved. This results in saturation of pass-through ratio to a maximum of 98% in PBSF even if a large number of banyans are used. Whereas if sufficiently large number of banyans are used in TBSF a pass-through ratio greater than 99.9% can be achieved.

The parallel tree banyan switch fabric (PTBSF) completely eliminates internal blocking and performs better than TBSF and PBSF with respect to switching delay and delay jitter. However this is true only for small size switches and also its hardware requirements are very large.

2.3 Conclusions

We have observed that although TBSF increases the passthrough ratio, latency also increases. MBSF decreases the latency factor but decreases the pass through ratio

as well. PBSF is a three dimensional switch that gives better performance both with respect to passthrough ratio as well as latency. However this is valid for only small size switches and we see that passthrough ratio saturates at 98%.

For the PTBSF switch architecture once a conflicting cell is routed down to next layer there is no mechanism for routing a cell to an upper layer back where an empty path may be available. One of the major drawback of 3-D based architectures is low utilization of the available hardware. There is no mechanism allowing cells which are routed down to make use of available free-paths in upper layers. We believe this factor significantly contributes in increasing hardware requirement under some level of performance. Also, most of the 3-D switches have complex switching elements such as 3×4 or 4×4 switching elements. Pipelined banyan is one of the simplest switch architectures. It exploits the spatial parallelism of the switch by parallelizing the data transmission with header routing. This switch compromises neither the latency nor throughput unlike TBSF and MBSF. In the next chapter we discuss the design and analysis of other switches based on this pipelined banyan architecture.

Chapter 3

Design and Analysis of Pipelined Switches

3.1 Introduction

Multi Stage Interconnection Networks (MIN's) have received significant interest among ATM switching researchers as a means of reducing the hardware complexity of the ATM switches. The concept of MIN was first introduced in the context of circuit switching. The aim was to design a non-blocking switch with less complexity than the crossbar switch. Apart from communication networks, MIN's have been well researched in the field of multiprocessor systems as a mechanism of interconnecting processors to processors and processors to memory modules. A number of MIN's have been proposed to-date varying in their complexity, throughput and cell loss ratio. Among these, a class of networks called delta networks became most attractive in ATM switching. A few examples of delta networks are Banyan, Baseline, Reverse Baseline, Omega, Modified Data Manipulator, Indirect binary-n-cube

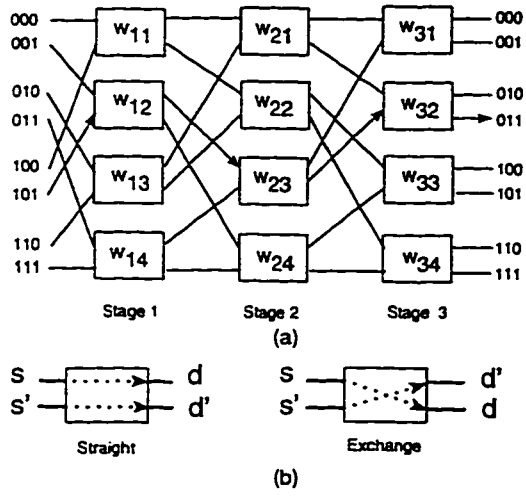

Figure 3.1: Simple banyan network

and Generalized Cube networks. These networks were shown to be topologically equivalent.

Banyan network belonging to the class of delta networks are among the simplest networks. A banyan switch based on omega network is shown in Figure 3.1. Some of the reasons for the popularity of banyan switches are as follows:

- Self - routing.
- Same latency for all input-output pairs.
- Have regular structure making them suitable for VLSI implementation.
- Complexity is O(NlogN)
- Unique path for each input-output thus preserving cell sequencing.

In this chapter we will discuss the design and analysis of a pipelined dilated switch.

3.2 Objective

An ATM switch should possess multipath feature to achieve high performance. Multipath networks allow multiple packets to reach a particular destination. Another feature which is required is that the performance of the switch should be scalable i.e. when the hardware resources are increased the performance should improve. From the survey of various switches shown in previous chapter we find that performance of some switches saturates at some level. For example the throughput of multi banyan switch saturates at 0.9 and the throughput of piled banyan saturates at 0.98. In these two switches the loss occurs anywhere inside the switch. Therefore switches with distributed loss are subjected to sautration in their performance. In switches such as tandem banyan switch and parallel tree banyan switch the loss occurs in last level of banyans. These switches with localized loss have a scalable performance. Therefore our objectives are as follows:

1. To design a localized loss switch architecture by using blocking switches with relatively simple hardware.

2. To use back-pressure mechanism and input buffering to reduce cell loss.

3. To maximize throughput by using parallel switch arrangement.

4. To minimize switching delay by sequential routing of headers and pipelining data transmission over switched paths.

With the above objectives in mind we design pipelined switch which has a control plane (for routing headers) and multiple data planes arranged in parallel (for transmitting cells). A pipelined dilated banyan switch has dilated banyans in control

plane and in data planes. In the following sections we first discuss the design of dilated banyan, switching elements used in dilated banyans followed by the pipelined switches.

3.3 Dilated Banyan (DB)

The idea behind dilated banyan network is to expand the internal link bandwidth to reduce packet loss probability to provide multiple paths. A dilated banyan (DB) with degree of dilation d is represented as $1 : 2^d$ DB. The degree of dilation gives the number of multiple outlets for each output port. An example of $1 : 2^2$ dilated banyan with 3 stages i.e. of size 8×8 is shown in Figure 3.2. This switch has $2^2 = 4$ links per output port. The first two stages of this switch are simply expansion stages which expand the input links in the form of a binary tree. The third stage has router phase. There is no cell loss in the first two stages. A cell loss can occur only in last stage. The dashed lines in the figure show the binary expansion of input I_0. The switching element in the router stage is explained in next section

A general structure of a $1 : 2^d$ DB is shown in Figure 3.3. As shown in the figure the DB has two phases; *Expansion Phase* and *Router Phase.* In the expansion phase the internal links are expanded in the form of a tree by doubling the links at each stage. In the router phase the number of links at the output of a stage remains the same as that of its input. Each stage of the expander consists of only demultiplexers (DM). For a $1 : 2^d$ DB with n stages there are d stages of *demultiplexers* in the expander and $(n - d)$ stages of *Dilated Switch Elements (D-SW)* in the router. In fact the demultiplexers in the expander are not arranged as shown in Figure 3.3. This figure only gives us an idea as to how demultiplexers and internal links

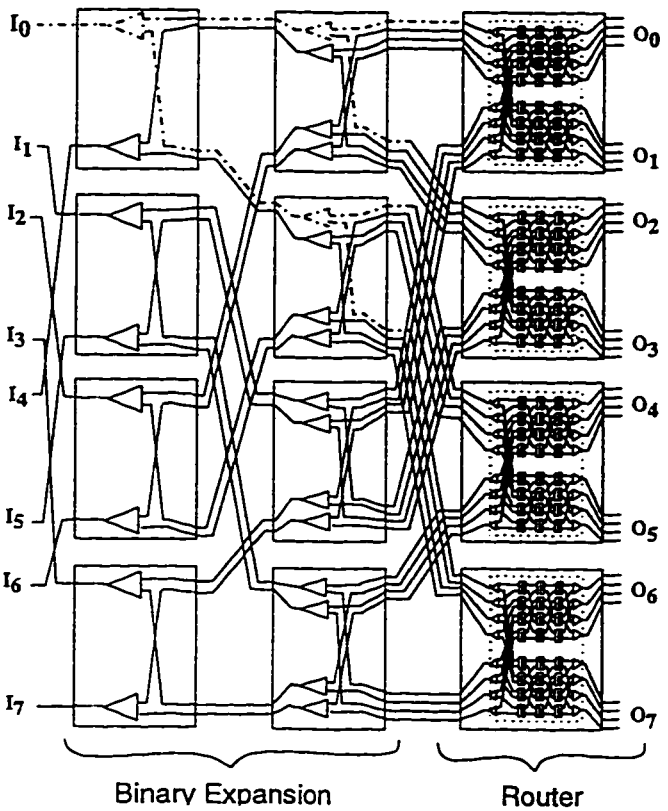

Figure 3.2: Dilated Switch (D-SW) for a $1 : 2^d$ dilated Banyan (DB)

grow in the form of a tree in the expander. Figure 3.4(a) shows the actual internal arrangement of the demultiplexers in a 2×2 *Expanded switching elements (E-SW)* in the expander for stages $0, 1 \ldots (d-1)$. Each DM here provides 2 output links for a cell at its input, one to upper output and the other to lower output. In between two stages, a group (or bundle) of links from a particular output are all permuted to the same input of the next stage. Since there are d stages in the expander, the last stage $(S_d - 1)$ has 2×2 switching elements with 2^d links per output port. The router has $(n - d)$ stages of *Dilated Switching Elements (D-SW)* as shown in the router of Figure 3.3. The internal architecture of a D-SW is not as simple as that of E-SW. In an E-SW the number of links at the output is twice that of the number

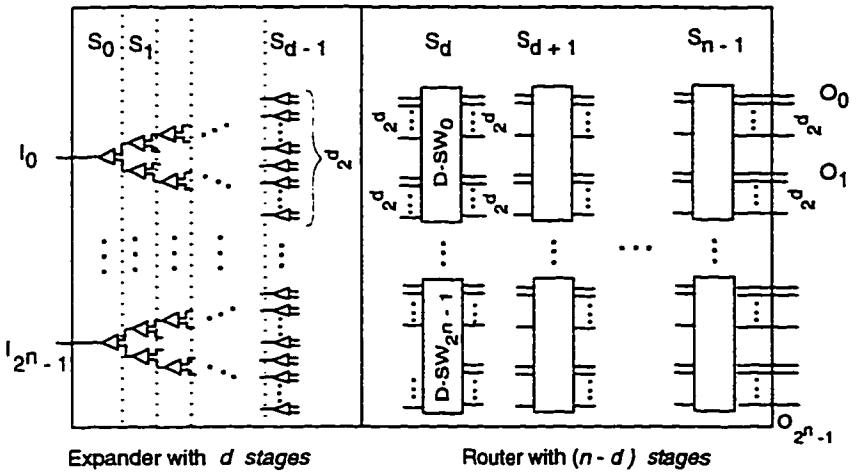

Figure 3.3: A 1:2^d dilated Banyan (DB)

of links at the input. Therefore there are sufficient links at the output to avoid any loss. However the D-SW has 2^d links per input port and 2^d links per output port as shown in Figure 3.3 router part. There can be at most 2^{d+1} cells at the input of a D-SW requesting the same output port, either the upper or the lower output. Since there are only 2^d links available at either the upper or the lower port, only 2^d cells will be routed successfully and the remaining cells will be dropped. There can be several internal architectures of the D-SW, designed in such a way that a loss occurs only if more than 2^d cells are requesting a particular output port. One possible architecture is crossbar. The *crossbar* although removes internal conflicts it does not provide multiple outlets. Also we want to route highest priority 2^d cells successfully which is not possible in crossbar. Therefore we have come up with another architecture based on the two-input *sorters* as shown in Figure 3.4(b).

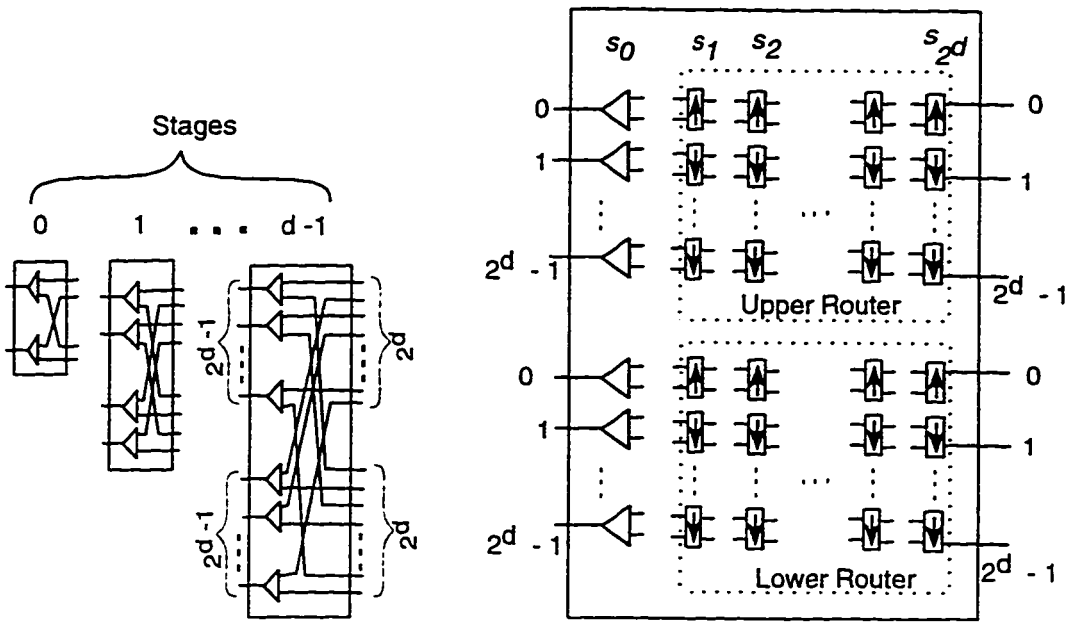

Figure 3.4: Switching elements for a $1 : 2^d$ dilated Banyan (DB)

3.3.1 Dilated Switch (D-SW)

D-WS is a switching element in the router phase of the dilated banyan. There are 2^{n-1} D-SW switches in each stage of the router phase. The D-SW switch in a $1 : 2^d$ dilated banyan has two groups of inputs and two groups of outputs with 2^d links per group. Cells can be examined for routing to either the upper output group or the lower output group. Figure 3.4 shows a *D-SW switching element* for a $1 : 2^d$ DB. It has 2^d input links and 2^d output links per port. The first stage of D-SW consists of demultiplexers. The demultiplexers double the total number of input links for both the input ports from 2^{d+1} to 2^{d+2}. From the second stage onwards we have 2-input sorters until the last stage. These sorters are symmetrical and

are divided into two groups called *upper router* and *lower router*. The term router from here onwards is used to refer to either upper router or lower router. The links between demultiplexer stage (s_0) and first stage of sorters (s_1) are arranged by a perfect shuffle permutation. This allows each cell at the input of a demultiplexer to be routed to either the upper router or the lower router on the basis of destination routing bit. There can be at most 2^{d+1} cells at the input of the first stage of the router. In the sorter stages the cells are routed on the basis of priority only. There are 2^d upper soters and 2^d lower sorters in each stage of the router. There are only two possible states of these sorters, staight and swap. The function of the upper and the lower sorters are as follows:

- An upper sorter (represented by upward arrow) routes the highest priority cell to its upper output if there are two cells. If there is only one cell it is routed to upper output regardless of the priority and the position of the cell at the input. In terms of state the sorter is set in staright state if there are two cells at the input and priority of cell at input 0 is greater than priority of cell at input 1, or if there is only one cell at input 0. In all other cases the sorter is set in swap state.

- A lower sorter (represented by downward arrow) routes the highest priority cell to lower output if there are two cells. If there is only one cell it is routed to lower output regardless of the priority and position of the cell at the input. In terms of state the sorter is set in staright state if there are two cells at the input and priority of cell at input 1 is greater than priority of cell at input 0, or if there is only one cell at input 1. In all other cases the sorter is set in swap state.

These upper and lower sorters are arranged as a banyan network with a perfect shuffle permutation between any two stages. The number of stages required in a D-SW of $1 : 2^d$ DB is equal to 2^d. This is proved in Theorem 1 under Complexity Analysis section 3.6. This arrangement of 2^d sorter stages allows the selection of the highest priority 2^d cells out of at most 2^{d+1} possible cells at the input of the first sorter stage. If there are less than or equal to 2^d cells, the sorters route all these cells to 2^d output links without any loss.

In Figure 3.4 we observe that there are 2^d sorters per stage (of the upper sorter or the lower sorter) and 2^d sorter stages. This gives $(2^d)^2$ sorters in the upper router and $(2^d)^2$ sorters in the lower router. Thus one sorter for a simple switch (Figure 3.6) has grown to $(2^d)^2$ sorters for a $1 : 2^d$ D-SW giving a complexity similar to a crossbar. However the sorter D-SW architecture still has advantages over crossbar as follows:

- Crossbar does not provide multiple outlets.
- Selecting cells on the basis of priority is not possible in crossbar.
- A small value of $d(< 3)$ has been shown to be sufficient to achieve very high throughput.

A dilated banyan will be completely non-blocking (both internally and externally) if the dilation degree is equal to the number of stages. However as we increase the dilation degree the complexity and the hardware resource requirements tend to grow very fast. Therefore we try to minimize the dilation degree by using some performance improvement techniques used in simple banyans. One of the best techniques recently proposed is pipelining the transmission of cells using several banyans planes

arranged in parallel [25]. This will be briefly discussed in the next section.

3.4 Pipelined Simple-Banyan

A detailed discussion of the pipelined simple-banyan has been given in the last section of literature review. Here, we will describe in brief the basic technique of pipelining and the advantages of pipelined simple-banyan.

A pipelined simple-banyan is constructed from a number of banyans arranged in parallel as shown in Figure 2.13. These banyans are called data planes. The routing decisions are performed in a separate banyan network called control plane. The difference between data planes and the control plane is that the data planes consist of combinational logic only, with routing paths set up by the control plane. Headers of the cells arriving at the input of the switch are applied to the control plane synchronously, at each time slot. Each time slot is divided into a number of reservation slots which is equal to the number of data planes present. In each reservation slot the control plane selects a data plane and routes the headers of the Head of line (HOL) cells and at the same time sets up the paths in this data plane for successfully routed headers. The unsuccessful HOL cells are attempted for routing in the next data plane during the next reservation slot. The control plane repeats this procedure for all the data planes in a round robin fashion. Once the paths are set in a data plane, cell transmission starts, while the control plane is busy performing routing for the next data plane. Thus reservation and cell transmission are performed in pipelined fashion. In the pipelined banyan paper [25] it has been shown that there is an upper limit to the number of reservation slots per time slot or in other words the number of data planes which can be placed in parallel. We

briefly recall this limit next by defining the following terms:

P = Packet size (= 424 bits for a 53 byte ATM cell)

T = Packet Transmission Time in Data Plane.

τ_s = Bit-time delay on input line (1 ns for a 1 Gbps line)

τ_c = Bit-time delay inside control plane (1 μs for a 1 Mbps)

l = Number of latency bits per stage in control plane.

n = Number of Stages.

K = Number of Data Planes.

t_c = reservation slot time (inside control plane).

t_s = input slot time.

From the above definitions we can derive t_c and t_s as follows:

$t_c = (n \times l + 1)\tau_c$ for a switch with n stages. The 1 is for the acknowledgement signal.

$t_s = P \times \tau_s$

The conditions for pipelining are :

1. $t_s > T$. This is because the data plane should be free for reservation by next time slot.

2. $t_s > K \times t_c$ to support K reservations in one time slot.

The second condition can be used to derive an upper limit on the number of data planes K as follows:

$$t_s \geq K t_c.$$

$$\Rightarrow P \times \tau_s \geq K(n \times l + 1)\tau_c.$$

$$\Rightarrow K < \frac{P \times \tau_s}{(n \times l + 1)\tau_c}.$$

The above equation can also be written as follow:

$$\tau_c \leq \tau_s \frac{P}{K(n \times l+1)}.$$

From the above equation we can say that the bit-time delay in the control plane is greater than the bit-time on the input line if $P \geq K(n \times l + 1)$. Therefore the clock rate of the control plane can be reduced thus allowing us to reduce the complexity of the control plane. The ratio $\frac{\tau_c}{\tau_s}$ gives the speed reduction of control plane over input line speed.

Advantages of Pipelined Simple-Banyan

- Parallelization of cell transmission with routing.
- Input buffering is useful unlike in single banyans where output buffering is shown to give better performance than input buffering.
- No routing issues in data planes since it has combination logic only.
- Speed up of control plane is achieved without serial-to-parallel conversion, a scheme usually used in other banyans to achieve speed increase.

3.5 Pipelined Dilated-Banyan (PDB)

Pipelined banyan uses a simple banyan in each data plane to reduce the cell loss ratio. However the throughput of a banyan network is very small ranging from 0.65 down to 0.2 for large size switches. Another banyan discussed in previous section is dilated banyan which gives very high throughput by dilating internal links. The cost paid in dilated banyan is large complexity due to link dilation. This complexity can be reduced to some extent by decreasing the dilation factor. But decreasing the

dilation factor will lead to reduction in throughput. The difference in throughput of dilated banyans and simple banyans is significant even for a small value of dilation factor. The throughput of dilated banyan can be improved by adopting some of the techniques used in simple banyans. One such technique used in simple banyans was input buffering. *Input buffering* places buffers at input ports. When a conflict occurs, one packet is forwarded while the other is kept back in the buffer. The input buffered switches were found to suffer from HOL blocking. In any given time slot, while a cell is waiting its turn for access to an output port, other cells may be blocked behind it despite the fact that their destination ports are possibly idle. Thus the switch can become congested if the traffic is bursty or the conflicts persist for a long time. For pipelined banyan, the problem is less severe because the service rate of pipelined banyan is higher than single banyan network. Due to multiple reservation slots, the input queues are evacuated at a much higher rate. The service rate can be increased further by using dilated banyans instead of simple banyans, since it has much higher throughput (even for small dilation factor) than simple banyan. As the complexity of dilated banyan is high, the service rate in pipelined dilated-banyan would be greater than in pipelined simple-banyan. To summarize a pipelined dilated-banyan has the following important features:

- Link dilation to increase the throughput.
- Pipelining to minimize the dilation factor.
- Input buffering to reduce the number of data planes.

The architecture of a pipelined dilated-banyan has buffers at the input ports, a control plane, and multiple data planes. An example of a dilated banyan of size

8 x 8 and dilation degree $d = 1$ is shown in Figure 3.5. Cells are generated at each time slot are stored in the input queues. The time slot is divided into a number of reservation slots equal to the number of data planes. In each time slot headers of the HOL cells in input queues are sent to the control plane. The control plane selects a data plane and performs reservation of paths for these HOL cells. This reservation of paths is done by the control plane for all data planes in round robin fashion.

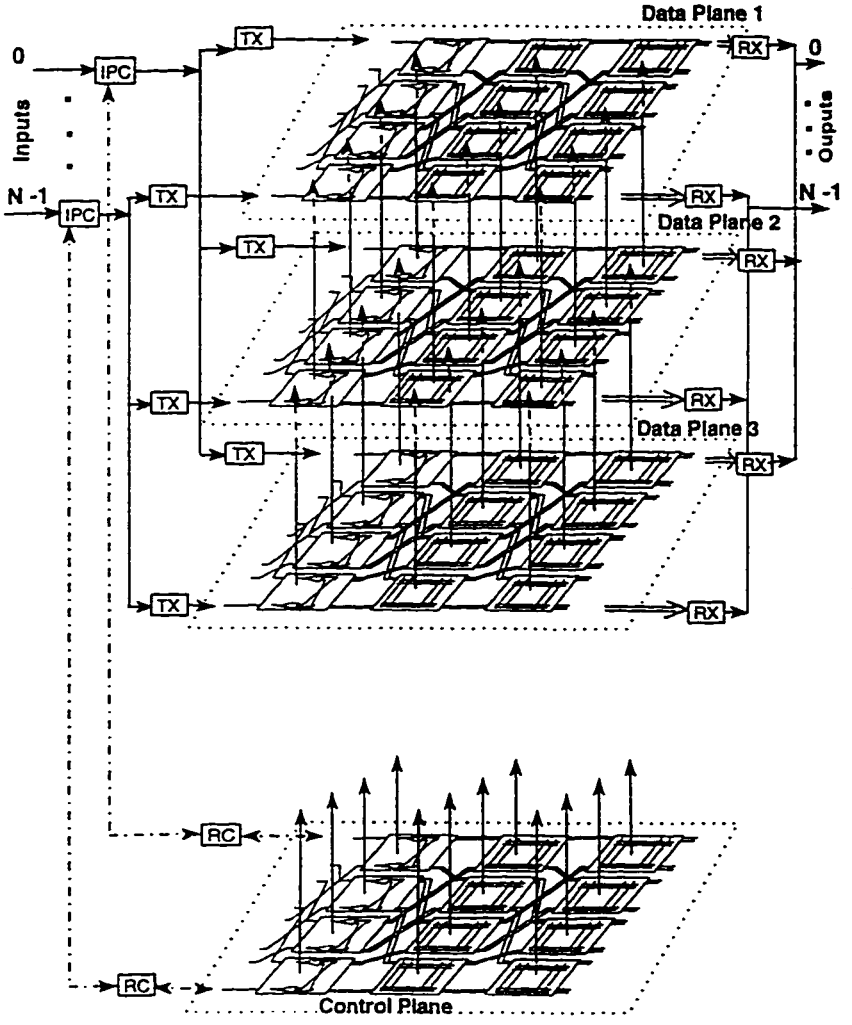

Figure 3.5: Example of 8 x 8 Switch with dilation degree = 1

3.6 Complexity Analysis

A number of switch architectures have been proposed to-date. Most of these switches were shown to be better than others with respect to some features. For example some switches were shown to have very high performance, but they also had very high hardware resource requirements. Some other switches such as TBSF, had high throughput (close to 100%), but its switching delay was relatively high. Thus, any switch designed has some cost associated with it. Complexity analysis is the evaluation of cost of a switch in terms of its hardware resource requirements and total switching delay performance.

Any space division switch based on MINs has the following hardware resource requirements.

- Switching Elements in each stage.
- Interconnection links between stages.
- Sometimes buffers at input and output ports of the switch.

In this section we evaluate the number of sorters, demultiplexers and interconnection links in a dilated banyan. We also present the total switching delay in dilated banyan.

Theorem 1: The Number of Stages in the Dilated Switch (D-SW) of a $1 : 2^d$ Dilated Banyan (DB) is equal to $2^d + 1$.

Proof:

A D-SW of a $1 : 2^d$ DB as has 2^d links per input/output group as described in section 3.2.1. Figure 3.6 shows examples of D-SW switches in $1 : 2^0$, $1 : 2^1$ and $1 : 2^2$ dilated banyans. Consider the $1 : 2^2$ D-SW in Figure 3.6(c). It has a demultiplexer

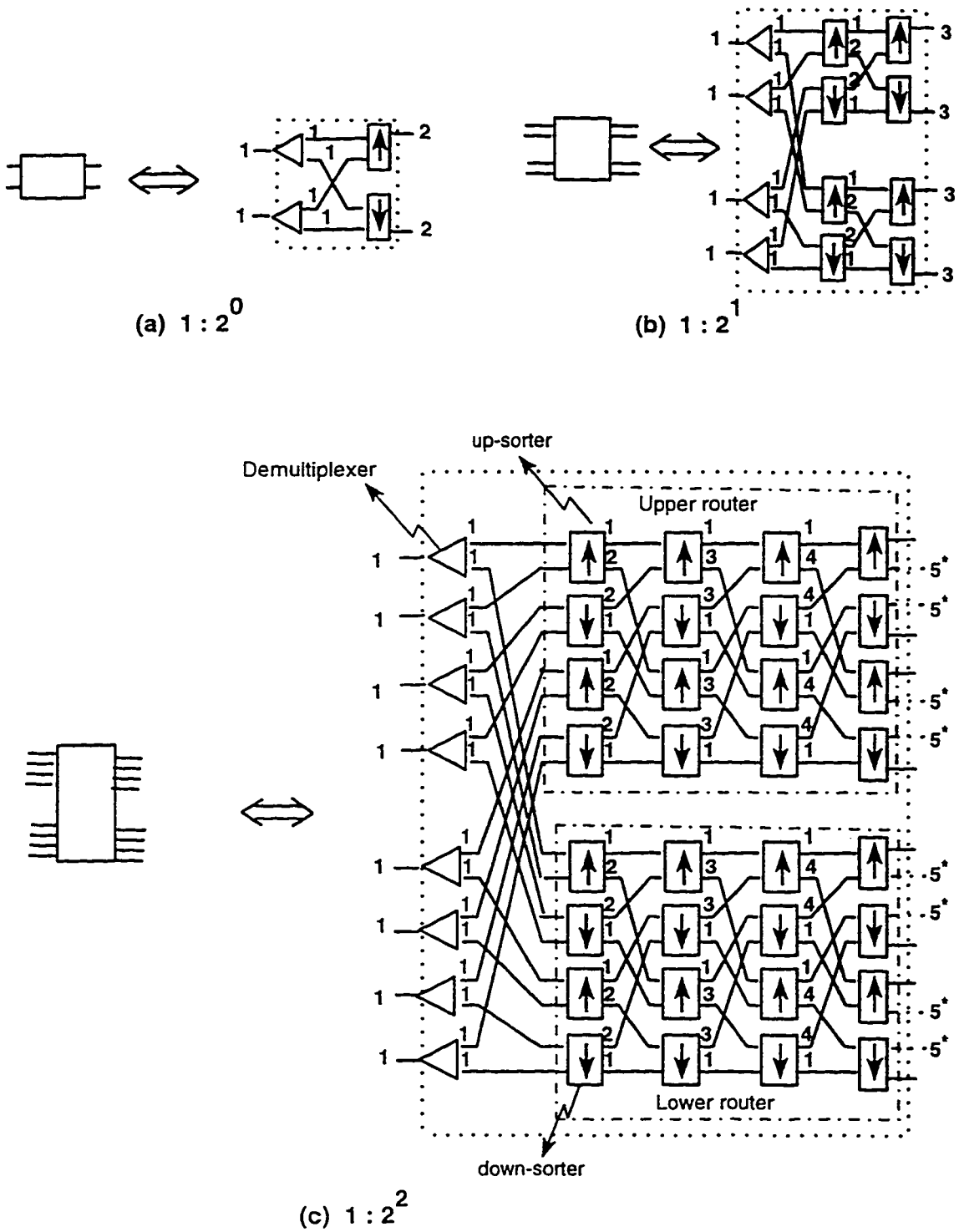

Figure 3.6: Examples of Dilated Switching Elements (D-SW).

stage followed by two groups of up/down sorter stages namely: *upper router* and *lower router*. The upper output of the up-sorter (in the router) has a cell if there is a cell at any one of its input. However the lower output of the up-sorter has a cell only if there are cells at both the inputs. Thus in up-sorter the upper output has an OR function and the lower output has an AND function. Since the down-sorter is symmetrical its upper output has an AND function and its lower output has an OR function.

A number on the link in Figure 3.6(c) represent the minimum number of cells that should be present at the input of the D-SW (with this link in their path to output group requested) for a cell to be present on this link. In sorter stage k the lower output of an up-sorter has a cell if there are atleast $k+1$ cells at the input of the D-SW. In D-SW with 2^d links per input group a cell is lost at its output only if there are more than 2^d cells at the input requesting for the same output group with 2^d links. The sorter stage 2^d has a cell at the lower output of up-sorter if there atleast 2^d+1 cells at the input of the D-SW. Since this is the minimum condition for a loss to occur in a D-SW of $1:2^d$ DB we stop at this sorter stage 2^d and take only the upper output of the up-sorter and the lower output of down-sorter. The other output link of the sorter (shown as dotted lines in Figure 3.6(c)) does not exist since a cell is present on this link if there are more than 2^d cells are present at the input of the D-SW requesting for this output group.

To conclude the D-SW of a $1:2^d$ DB has 2^d+1 stages (1 Demultiplexer stage + 2^d sorter stages).

Theorem 2: A $1:2^d$ dilated banyan with n stages has $(n-d)2^{n+2d}$ Sorters and $2^{n+d}(n-d+1)-2^n$ Demultiplexers.

Proof:

Number of Sorter stages per D-SW = $N_{Sorter\ stages}^{D-SW} = 2^d$

Number of sorters in each stage in D-SW = $N_{Sorters}^{D-SW\,stage} = 2^{d+1}$

Number of sorters in each D-SW:

$$\begin{aligned} N_{Sorters}^{D-SW} &= N_{Sorters}^{D-SW\ stage} \times N_{Sorter\ stages}^{D-SW} \\ &= 2^{d+1} \times 2^d \\ &= 2^{2d+1} \end{aligned}$$

Total number of sorters in a $1 : 2^d$ DB with n stages = $N_{Sorters}^{DB}$

$$\begin{aligned} N_{Sorters}^{DB} &= N_{Sorters}^{D-SW} \times N_{D-SW}^{Router\ stage} \times N_{D-SW\ stages}^{Router} \\ &= 2^{2d+1} \times 2^{n-1} \times (n-d) \\ &= (n-d)2^{n+2d} \end{aligned} \tag{3.1}$$

Number of Demultiplexers in Expander = $N_{DMUX}^{Expander}$

$$\begin{aligned} N_{DMUX}^{Expander} &= 2^n + 2^{n+1} + \ldots + 2^{n+d-1} \\ &= 2^n(2^d - 1) \end{aligned}$$

Number of Demultiplexers in Router = N_{DMUX}^{Router}

$$\begin{aligned} N_{DMUX}^{Router} &= N_{DMUX}^{D-SW} \times N_{D-SW}^{(Router\ stage)} \times N_{(D-SW\ stages)}^{Router} \\ &= 2^{d+1} \times 2^{n-1} \times (n-d) \\ &= (n-d)2^{n+d} \end{aligned}$$

Total number of Demultiplexers in a $1:2^d$ DB with n stages $= N_{DMUX}^{DB}$

$$\begin{aligned} N_{DMUX}^{DB} &= N_{DMUX}^{Expander} + N_{DMUX}^{Router} \\ &= 2^n(2^d-1) + (n-d)2^{n+d} \\ &= 2^{n+d}(n-d+1) - 2^n \end{aligned} \tag{3.2}$$

Theorem 3: A $1:2^d$ dilated banyan with n stages has $\{2^n(n2^d - d2^d + 2^d - 1)\} + \{(n-d)2^{n+2d+1}\}$ Interconnection links (IC).

Proof:

Number of Interconnection links in Expander $= N_{IC}^{Expander}$

$$\begin{aligned} N_{IC}^{Expander} &= 2^n + 2^{n+1} + 2^{n+2} + \ldots + 2^{n+d} \\ &= 2^{n+d+1} - 2^n \end{aligned}$$

Number of Interconnection links in each D-SW $= N_{IC}^{D-SW}$

$$\begin{aligned} N_{IC}^{D-SW} &= 2^{d+2} \times (2^d + 1 - 1) \\ &= 2^{2d+2} \end{aligned}$$

Number of Interconnection links in Router $= N_{IC}^{Router}$

$$\begin{aligned} N_{IC}^{Router} &= \{N_{IC}^{D-SW} \times N_{D-SW}^{S} \times N_{D-SW\ stages}^{Router}\} + \\ &\quad \{N_{IC}^{Between\ D-SW\ Stages} \times (N_{D-SW\ stages}^{Router} - 1)\} \\ &= \{2^{2d+2} \times 2^{n-1} \times (n-d)\} + \{2^{n+d} \times (n-d-1)\} \\ &= \{(n-d)2^{n+2d+1}\} + \{(n-d-1)2^{n+d}\} \end{aligned}$$

Total number of IC in a $1:2^d$ DB with n stages = N_{IC}^{DB}

$$\begin{aligned} N_{IC}^{DB} &= N_{IC}^{Expander} + N_{IC}^{Router} \\ &= \{2^{n+d+1} - 2^n + (n-d-1)2^{n+d}\} + \{(n-d)2^{n+2d+1}\} \\ &= \{2^n(n2^d - d2^d + 2^d - 1)\} + \{(n-d)2^{n+2d+1}\} \end{aligned} \tag{3.3}$$

Theorem 4: Switching Delay along a path from inputs to output of a $1:2^d$ dilated banyan of size n stages is $\{3n + 10(n-d)2^d\} \times \tau_{gate}$ where τ_{gate} is delay in a single gate.

Proof:

Total number of stages in $1:2^d$ DB = N_{Stages}^{DB}

$$\begin{aligned} N_{Stages}^{DB} &= N_{Stages}^{Expander} + N_{Stages}^{Router} \\ &= dDMUX\ stages\ +\ (n-d)(2^d\ Sorter\ stages\ +\ 1\ DMUX\ stage) \\ &= n\ DMUX\ stages\ +(n-d)2^d\ Sorter\ stages \end{aligned}$$

A sorter in control plane has a memory element, D flip flop and multiplexers. The memory element has a state function with 4 variables of form $F(pr_0, pre_0, Pr_1, pre_1,)$ where pr_0, pr_1 are priority bits and pre_0, pre_1 are presence bits of the incoming two cells. If dual input gates are used the number of gate levels required to implement a sum of product form of this function is $log_2(2^{n_v} \times n_v)$ where n_v is number of boolean variables per minterm.

Number of gate levels in a state with 4 variables = $log_2(2^4 \times 4) \approx 6$.

Number of gate levels in a D flip flop = 2.

Number of gate levels in a Muliplexer = 2.

Number of gate levels in a Dmultiplexer = 3.

Total number of gate levels in the sorter = 6 + 2 + 2 = 10.

Let τ_{DMUX} be the delay in one Demultiplexer stage,

τ_{Sorter} be the delay in one Sorter stage.

Total switching delay in a $1:2^d$ DB with n stages = T_{DB}

$$\begin{aligned} T_{DB} &= n \times \tau_{DMUX} + (n-d)2^d \times \tau_{Sorter} \\ T_{DB} &= \{3n + 10(n-d)2^d\} \times \tau_{gate} \end{aligned} \tag{3.4}$$

where τ_{gate} represents delay in a single gate.

Theorem 5: An Expanded Banyan Switching Fabric(EBSF) with n stages, Expansion Factor EF and $e = log_2 EF$ has $2^{n+e}(n-e)$ sorters, $2^{n+e}(n-e+1) - 2^n$ demultiplexers, $2^{n+e}(3n-3e+1) - 2^n$ Interconnections(IC) and $13n \times \tau_{gate}$ switching delay in terms of gate levels, where τ_{gate} represents a single gate delay.

Proof:

An $(N \times N)$ EBSF has $EF(N \times N)$ interleaved delta Networks [7]. This switch has two parts similar to dilated banyan, an expansion part and a routing part. The expansion part has only demultiplexers. The router part hs 2×2 switching elements with dilation factor 0 i.e. they are $1:2^0$ D-SW as shown in Figure 3.6(a). The first e stages are in expansion part followed by $(n-e)$ stages in router part.

The number of Demultiplexers in Expansion part = $N_{DMUX}^{Expander}$

$$\begin{aligned} N_{DMUX}^{Expander} &= 2^n + 2^{n+1} + \ldots + 2^{n+e-1} \\ &= 2^n(2^e - 1) \end{aligned}$$

The number of Demultiplexers in Router part = N_{DMUX}^{Router}

$$\begin{aligned} N_{DMUX}Router &= 2^{n+e-1} \times 2 \times (n-e) \\ &= 2^{n+e}(n-e) \end{aligned}$$

The total Number of Demultiplexers in an EBSF = N_{DMUX}^{EBSF}

$$\begin{aligned} N_{DMUX}^{EBSF} &= N_{DMUX}^{Expander} + N_{DMUX}^{Router} \\ &= 2^n(2^e - 1) + 2^{n+e}(n-e) \\ &= 2^{n+e}(n-e+1) - 2^n \end{aligned} \tag{3.5}$$

The total number of Sorters in an EBSF = $N_{Sorters}^{EBSF}$

$$\begin{aligned} N_{Sorters}^{EBSF} &= 2^{n+e-1} \times 2 \times (n-e) \\ &= 2^{n+e}(n-e) \end{aligned} \tag{3.6}$$

The number of Interconnection links in Expansion part = $N_{IC}^{Expander}$

$$N_{IC}^{Expander} = 2^n + 2^{n+1} + \ldots + 2^{n+e} = 2^n(2^{e+1} - 1)$$

The number of Interconnection links in Router part = N_{IC}^{Router}

$$\begin{aligned} N_{IC}^{Router} &= 4 \times 2^{n+e-1} \times (n-e) + 2^{n+e} \times (n-e-1) \\ &= 2^{n+e}(3n - 3e - 1) \end{aligned}$$

The total Number of Interconnection links in an EBSF $= N_{IC}^{EBSF}$

$$\begin{aligned} N_{IC}^{EBSF} &= N_{IC}^{Expander} + N_{IC}^{Router} \\ &= 2^n(2^{e+1} - 1) + 2^{n+e}(3n - 3e - 1) \\ &= 2^{n+e}(3n - 3e + 1) - 2^n \end{aligned} \tag{3.7}$$

The total switching delay in an EBSF $= T_{EBSF}$.

$$\begin{aligned} T_{EBSF} &= n \times \tau_{DMUX} + n \times \tau_{Sorter} \\ &= 3n \times \tau gate + 10n \times \tau gate \\ &= 13n \times \tau_{gate} \end{aligned} \tag{3.8}$$

3.7 Comparisons

In the last section we have evaluated the hardware resources and delay of a single plane banyan switch. Here we will give a comparison of hardware resource requirements and total switching delay for various multi-plane banyan switches. It should be observed that the results derived in the last section for a dilated banyan can also be used for a simple banyan, since a simple banyan is a special case of a dilated banyan with dilation factor $d = 0$. For comparison purpose, the 2×2 switching elements of a simple banyan is represented using demultiplexers and sorters as shown in Figure 3.6(a). In table 3.1 we show the results derived in previous theorems. From this table we observe that the resource reuirements of dilated banyan is of the order of square as compared to simple banyan.

Hardware Resources	Simple banyan(SB)	$1:2^d$ Dilated banyan(DB)	EF times Expanded banyan(EBSF)
$N_{Sorters}$	$n2^n$	$(n-d)2^{n+2d}$	$2^{n+e}(n-e)$
N_{DMUX}	$n2^n$	$2^{n+d}(n-d+1)-2^n$	$2^{n+e}(n-e+1)-2^n$
N_{IC}	$3n2^n$	$\{2^n(n2^d-d2^d+2^d-1)\}$ $+\{(n-d)2^{n+2d+1}\}$	$2^{n+e}(3n-3e+1)-2^n$
T_{SB}	$13n \times \tau_{gate}$	$\{3n+10(n-d)2^d\} \times \tau_{gate}$	$13n \times \tau_{gate}$

Table 3.1: Hardware resources required in single banyan switches

Switch Name	Number of Sorters	Number of Demultiplexers	Number of Inter-connection links	Delay in gate levels	Cell loss Probability
SB	10240	10240	30720	130	$7.6e \times 10^{-1}$
DB_2	36864	19456	93184	226	3.6×10^{-1}
DB_4	131072	35840	297984	442	2.2×10^{-2}
DB_8	458752	64512	982016	874	4.7×10^{-6}
$EBSF_2$	18432	19456	56320	130	5.5×10^{-1}
$EBSF_4$	32768	35840	101376	130	3.4×10^{-1}
$EBSF_8$	57344	64512	179200	130	2.0×10^{-1}
$EBSF_{512}$	524288	1047552	2096128	130	4.9×10^{-4}

Table 3.2: Comparison of Sorters, DMux, IC Single Banyan Switches

Table 3.2 gives the hardware resources required and performance achieved for a 1024 x 1024 switch. In this table the first row for simple banyan(SB) has the least performance. Rows 2,3 and 4 show hardware requirements of dilated banyan(DB) with dilation degree 1,2 and 3. With DB, very low cell loss probability (CLP) can be achieved by increasing the dilation degree. However as seen in the table the cost of the DB switch is very high in terms of number number of Sorters, Demultiplexers and Interconnection links for higher dilation degree. The last four rows show the requirements of expanded banyan switch fabric (EBSF). In $EBSF_{EF}$, the subscript EF represents the expansion factor, which is equal to the number of banyans inter-

Switch Name	K	Number of Sorters	Number of Demultiplexers	Number of Interconnection links	Delay in gate levels	Cell loss Probability
PSB	6	61440	61440	184320	780	1.4×10^{-6}
PDB_2	2	73728	38912	186368	452	1.3×10^{-6}
$PEBSF_2$	3	55296	58368	168960	390	5.6×10^{-6}
$PEBSF_4$	2	65536	71680	202752	260	1.2×10^{-6}
$TBSF$	14	143360	143360	430080	1820	1×10^{-6}
$PTBSF$	19	194560	194560	583680	208	1×10^{-6}

Table 3.3: Comparison Table for Sorters, DMux, IC and Delay in various pipelined and replicated switches

leaved to obtain this expanded switch. The three rows of DB_D and $EBSF_{EF}$ show that for the same value of D and EF the number of sorters and interconnection links in an $EBSF$ is less than that in a DB. However the CLP for the EBSF is very high. To achieve a CLP close to 10^{-6}, an EBSF requires 512 banyans to be interleaved as shown in the last row. The number of sorters and interconnection links required by this $EBSF_{512}$ is very high. In the above table, the fifth column presents the delay in various banyans. We can see that the performance of dilated banyans can be increased by increasing the dilation degree. However the delay also increases with an increase in dilation degree. In pipelined expanded-banyan the delay would always be the same whatever would be the expansion factor. Table 3.3 gives the comparison of multi-plane banyans for a switch of size 1024 x 1024. Since the advantage of pipelined banyans is only with input buffering, we present here the results of input buffered switches with input queue size 10. The cell loss values actually correspond to the last figure in next chapter where we have presented the performance comparision of different pipelined banyans under uniform traffic. For the purpose of comparison we have shown the requirements of various switches for a CLP of 10^{-6} . From the table we can see that the pipelined simple-banyan requires

higher number of data planes, which leads to larger delay in control plane. The second row for pipelined dilated-banyan shows that only 2 data planes are required to achieve cell loss around 10^{-6}. The number of sorters and interconnection links required in PDB_2 is almost the same as in PSB. The third row shows the results of pipelined expanded-banyan with expansion factor 2. $PEBSF_2$ hardware resource requirements are in between PSB and PDB_2, while the delay is less than both. The fourth row shows the results of pipelined expanded-banyan with expansion factor 4. The hardware resource requirements of this switch are close to PSB and PDB_2. But the delay in its control plane is the least among all four types of pipelined switches. In the last two rows we show the hardware resource requirements of a TBSF and PTBSF switches. As seen here the TBSF requires large number of hardware resources and also it has highest delay characteristic. The PTBSF has the least delay among all the switches shown in this delay but its hardware resource requirement is greater than the TBSF switch.

3.8 Conclusion

In this chapter we have discussed the design of a pipelined dilated-banyan. The pipelined dilated-banyan has three important features: link dilation, pipelining and input buffering. We have stated theorems giving hardware resources required in a dilated banyan and expanded banyans. In the end we have compared the pipelined dilated-banyan with various switches. In the next chapter we will discuss the performance of various pipelined banyans under uniform traffic sources.

Chapter 4

Performance Analysis

4.1 Introduction

Space division switches are self routing, have distributed control and provide multiple concurrent paths between inputs and outputs. However due to limited resources available ia a switch fabric it may not be possible to set all the required paths simultaneously. This characteristic is referred to as *blocking*. The are two types of blocking possible, external blocking and internal blocking. If two packets request the same output port it is called *external blocking*. For example in a crossbar switch fabric all incoming packets can reach their respective destinations as long as there is no output conflict. If on the other hand, there is more than one packet in the same slot destined to the same output port, then only one packet can be routed to that output. The remaining packets will have to be either dropped or buffered somewhere. An event of *internal blocking* occurs when more than one packet at the input of a switching element request the same link inside the switch fabric. Banyan interconnection networks suffer from both internal and external blocking, while the

crossbar switch suffers only from external blocking. An internal blocking may occur even if the two packets are destined for two distinct output ports. The existence of these conflicts results in the limitation of maximum achievable throughput of the switch. The effect of conflicts increases with an increase in the switch size. In this chapter we present the performance analysis of pipelined banyans both by analytical and simulation methods.

4.2 Performance Issues

All the switch architectures based on banyan networks include means to overcome blocking and improve performance. The performance issues of concern for an ATM switch fabric are throughput or cell loss rate, switching delay, delay jitter, reliability and fault tolerance. *Throughput* ($TP(p)$) is defined as the average number of cells which are successfully delivered by the switch per time slot per input line at input load of p for $0 \leq p \leq 1$. Maximum throughput is $p = 1$, and normalized throughput is defined as $TP(p)/p$. *Cell loss probability (CLP)* is defined as the fraction of cells lost as a result of blocking and/or buffer overflows. In ATM networks cell retransmission takes place on an end-to-end basis because of the high speeds involved in these networks. Therefore loss is one of the most important performance measures. *Switching Delay* is defined as the average time a cell spends from the time it arrived at an input port till it is successfully delivered on its requested output port. This includes the time spent in any input, internal and output buffers. For a network user, the end-to-end delay is an important factor which is the sum of the delay of individual switching nodes on the path. *Delay jitter* is defined as the time difference of two cells in the same time slot to reach their destination. Large delay jitter creates

cell sequencing problem.

The factors which influence the above performance aspects of the switch fabric are:

- Input load.
- Buffering at input ports, output ports, and at switching elements inside the switch fabric.
- Traffic pattern according to which packets arrive at inputs of a switch.

Buffering Strategies

Based on the location of the buffers, ATM switches can be input-buffered, internally-buffered, output-buffered, or any combination of these. Consider the example of a non-blocking switch which can clear all incoming cells to the output side of the switch before a new time slot begins. Assuming that all input and output lines operate at the same speeds if two or more cells request the same output port in the same time slot, the output line can service only one cell per time slot. The other cells must be buffered. This type of buffering is referred to as *output buffering* since buffers are physically placed at the output side of the switch. In *input buffered* switches, buffers are placed at the input ports to save any undelivered cells. The third type of buffering strategy is internal buffering whereby buffers are placed within the switch fabric at possible conflicting points. This type of buffering is usually not preferred due to the following reasons:

- Cells may be delivered out of sequence in multipath switches.
- It complicates the internal design of switching the elements.
- It complicates the fault diagnosis testing [34].

Input buffering is generally not effective if the throughput of the switch fabric is low. In input buffered switches, arriving cells are stored in a First-in-First-out (FIFO) queues. These FIFO queues cause HOL (Head of the Line) blocking. In HOL blocking the undelivered HOL cells may block the cells behind them even if their destination ports are possibly idle. If the conflicts persist for some time, the input queues overflow. Therefore, input buffering is a useful technique only if effective measures are taken first to achieve high throughput for the switch. It is shown in [34] that output buffering significantly outperforms input buffering with regard to the throughput-delay performance under uniform traffic. However output buffered switches are usually more complex because the output memories are not easy to implement. Each output memory must have a minimum bandwidth of $(N+1)v$ (where v is the speed of external lines), corresponding to a maximum of N write and a single read operations [30]. On the other hand memory speed does not constitute a major concern for input buffered switches.

Input Traffic Pattern

Traffic as seen by the input ports of the switch is described by two random processes:

- The process that describes the arrival of packets at the inputs of the switch
- The process which describes the destination request distribution for arriving packets.

Some of the input traffic models used for the performance evaluation of ATM traffic are described below:

Uniform traffic: This is one of the simplest traffic patterns. Cells arrive at an input line of the switch according to IIR(independent and identically distributed random) Bernoulli process, with parameter $p(0 < p \leq 1)$, independent from all

other input lines. Here p represents the input load or the arrival rate of a particular input. An incoming cell chooses its destination uniformly among all N output ports and independently from all other requests. This traffic model is also referred to as independent uniform traffic or simply random traffic model. Uniform traffic is one of the most widely used input traffic model to evaluate the performance of switch fabrics for the following reasons:

- This assumption makes the analytical evaluation of the switch more easy.
- The implementation of simulator is easy since a random number generator function can be used to generate the incoming traffic.

Bursty Traffic: In this model cells arriving at the switch input ports exhibit strong correlation in the output ports requested. The correlation may occur in time domain or space domain or in both. The correlation is in time domain if cells arriving on a given line exhibit correlated output port requests and traffic on different lines are not correlated. In bursty traffic the correlation is in time domain with cells arriving in the form of bursts of random length for short duration. Bursty traffic is defined in terms of burst length, the gap between consecutive bursts and the process describing the output ports requests for bursts.

Permutation traffic model: In this model the arriving cells exhibit strong correlation in space domain, whereby cells arriving on various input lines in a given time slot have correlated output port requests, but cells arriving on different slots are not correlated. This is called permutation traffic because, the destinations requested by cells in a given time slot constitute a permutation of set $(1, 2, \ldots N)$. Since all destination requests are unique there is no output conflict. Since there is no output conflict, crossbar switches can pass this type of traffic with no loss.

Output Concentration traffic: This model has output port requests correlated in space domain as described above. Traffic entering the switch on all N input lines is destined to only a subset of all output lines.

Communities of Interest: This model shows output port request correlation in both time domain and space domain. The input traffic generates a specific output request pattern such that the limitation on throughput is primarily due to contention on internal lines. The limitation may be incurred even if there is no output port conflict.

4.3 Assumptions for Analytical and Simulation models

The switch architectures described below , dilated banyan, pipelined simple banyan. pipelined dilated banyan and pipelined expanded banyan are analyzed under the following assumptions.

- The size of all arriving cells is fixed.
- Arrival of packets at the switch inputs are independent and identical Bernoulli distributed.
- Packets are directed with the same probability to all outputs.
- All the switches in the network are synchronized by a single clock.
- There can be only one packet on a particular internal link during each time slot.

- In case of internal conflicts in unbuffered switches, only one cell is routed successfully and remaining cells (more than one in case of dilated banyans) are discarded i.e. they are not submitted at a later time.
- The switches and links have no internal buffers to temporarily store an incoming packet that cannot be forwarded in the current cycle.

4.4 Dilated Banyan

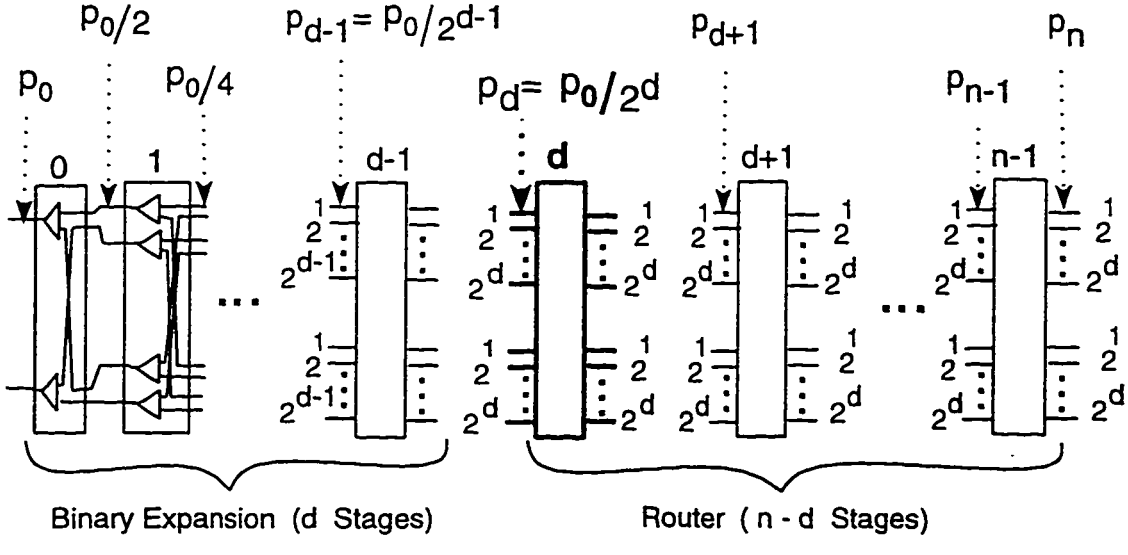

Figure 4.1: Load at various stages in a dilated banyan

4.4.1 Analytical Model

A dilated banyan as discussed in section 3.2 has an expansion part and a router part. For a switch of size $N \times N$ and dilation degree d there are $n = log_2 N$ stages and 2^d links per output port. The first d stages are in the expansion part and remaining $(n - d)$ stages are in the router part.

Expander:

The interconnection links are expanded as binary tree in the expander. The expander has only demultiplexers in each stage. The demultiplexers double each input link as shown in Figure 4.1 with one link going to upper output and other going to lower output of the 2×2 SE. Thus a cell can exit at either of the outputs. Since the number of output links is twice that of number of input links there is no loss of cells in the expansion part. A cell can appear successfully either at the upper output or the lower output depending on the destination bit. Therefore, the probability that a cell appears successfully at an output link is half the probability of a cell appearing at the input link.

Let P[Cell appears on an input link] $= p_0$

P[Cell appears on a link at input of stage i] $= p_i = p_0/2^i$ for $(0 \leq i \leq d)$

Therefore $p_d = 2^{-d}$ represents the load at input of stage d which is the first stage of the router part after 0 to $d-1$ stages in the expansion part.

Router:

The router has D-SW switching elements as described in section 3.3.1. In the expander due to binary expansion of links at each stage there is no cell loss. However in router there is no expansion of links at each stage. The D-SW switching element in the router has same number of links at its input and output. Therefore the probability of a cell appearing successfully at the output of stage d depends on the number of cells at the input. As seen in Figure 4.1 a D-SW has two groups of inputs and outputs with 2^d links in each group. In the worst case there can be 2^{d+1} cells. all requesting the 2^d links of the upper output group. Since there are only 2^d links in the output group atmost 2^d cells will be routed successfully and the remaining

cells are lost.

The passthrough for stages d to $(n-d)$ of the router can be evaluated as follows:

We know that P[Cell on a link at the input of stage d] $= p_d = 1/2^d$

Let P[Cell on a link at the input of stage i] $= p_i$ for $d \leq i \leq n-1$

P[m cells are present at inputs of a D-SW] $= p_i^m(1-p_i)^{2^{d+1}-m}\binom{2^{d+1}}{m}$

P[At most 2^d cells are selected out of $m = 0$ *to* 2^{d+1} cells] $= P_B = \sum_{j=1}^{m} \frac{min(j,2^d)}{2^d}\binom{m}{j}$

P[m most prior cells are selected at an output group] $= P_C = P_A \times P_B$

$$= \binom{2^{d+1}}{m} p_i^m (1-p_i)^{2^{d+1}-m} \times \sum_{j=1}^{m} \frac{min(j,2^d)}{2^d}\binom{m}{j}$$

Let p_m = P[There is a cell on a link at the output group] $= \frac{1}{2^m} \times P_C$

$$= \frac{1}{2^m}\binom{2^{d+1}}{m} p_i^m (1-p_i)^{2^{d+1}-m} \times \sum_{j=1}^{m} \frac{min(j,2^d)}{2^d}\binom{m}{j}$$

The Probability that there is a cell on any link at the output of a D-SW switch in stage i is given by:

$$p_{i+1} = \sum_{m=1}^{2^{d+1}} p_m \qquad for\ i = d, d+1, \cdots, n-1 \tag{4.1}$$

$$where \quad p_m = \frac{1}{2^m}\binom{2^{d+1}}{m} p_i^m (1-p_i)^{2^{d+1}-m} \times \sum_{j=1}^{m} \frac{min(j,2^d)}{2^d}\binom{m}{j}. \tag{4.2}$$

p_m is the probability that at most 2^d or m (whichever is minimum) are routed successfully out of m cells destined for an output group. Solving Equation 4.2 for p_n recursively we obtain the probability of a cell appearing successfully on any one link at the output. Since there are 2^d links per output port the passthrough of dilated banyan would be $2^d \times p_n$ and the cell loss probability would be $1 - 2^d \times p_n$.

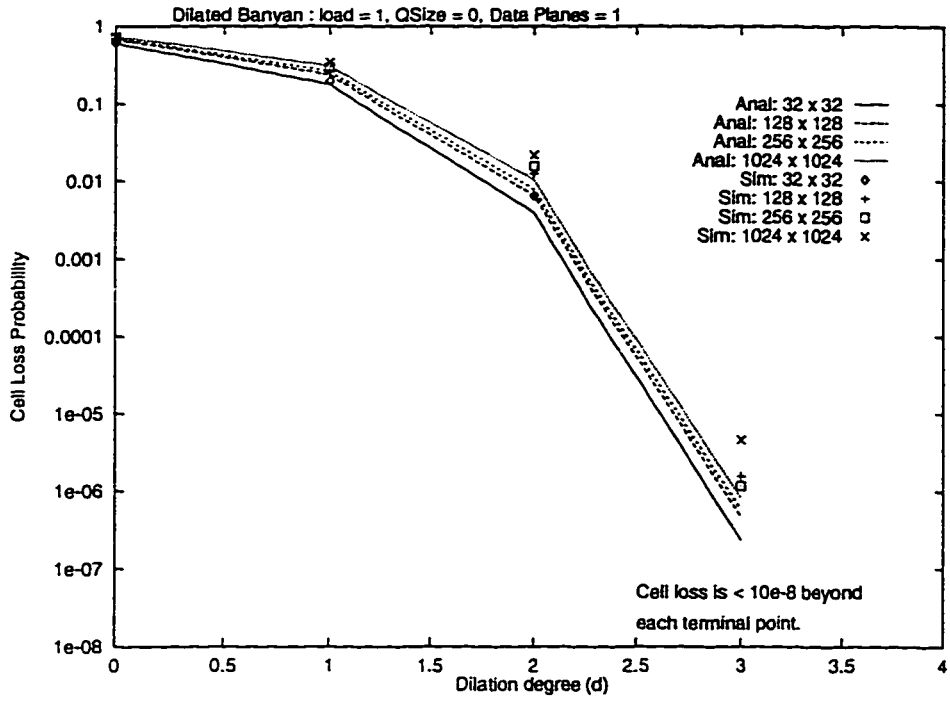

Figure 4.2: Cell loss probability in unbuffered dilated banyans at full load.

4.4.2 Simulation

Figure 4.2 shows the analytical and simulation results for an unbuffered dilated banyan. The simulations were performed under full load with uniform traffic. The plot shows cell loss probability for dilated banyans with dilation degree $d = 1, 2, 3 and 4$. Dilated banyan with dilation degree d has 2^d output links per output port. In the plot, lines represent analytical results and symbols represent simulations results. The analytical results are more optimistic than simulations with a maximum error of 10^{-1}. From the simulations we did not observe any cell loss for dilation degree 4 for a simulated load of 10^8 cells. Therefore, we could safely conclude that CLP $< 10^{-8}$ for a dilation degree of 4 and switch size as large as 1024 x 1024. For example consider a switch of size 32 x 32. Suppose that the simulator is run for 10^6 iterations. Under full load 32 x 10^6 cells would be generated, and in the worst case

only one cell would be lost. Then in this situation the cell loss probability would be 3.125×10^{-8}. Thus the upper limit on the precision of cell loss probability that can be obtained by simulations is limited by the number of simulation runs. From Figure 4.2 it can be concluded that in a switch as large as 1024×1024, a dilation degree of 4 is sufficient to achieve a cell loss probability of 10^{-8} under full load. However the complexity of 1024×1024 with dilation degree 4 would be very high. The Performance of buffered dilated banyan is shown in Figure 4.3. This figure shows the cell loss probability for switches of various sizes with input buffer size 5. All simulations for buffered dilated banyans were performed for 10^4 iterations. Therefore the precision of CLP which could be obtained is between 3.13×10^{-6} (for 32×32 switch) to 9.77×10^{-8} (for 1024×1024 switch). Figure 4.3 shows that to achieve a CLP around 10^{-6} a dilation degree 3 is sufficient with input buffer size 5. Figure 4.4 illustrates the effect of increasing the buffer size on a 32×32 switch at full load. The input buffer size was varied from 1 to 9. It can be seen that there is no improvement in CLP with an increase in buffer size at input ports. This is because at full load there is a continuous inflow of cells at each time slot. However due to internal conflicts some cells remain at the head of the line and block other cells behind them. This process called head of the line blocking causes the queues to overflow after some time. It was observed in simulations that at full load the queues fill after few iterations whatever may be the input queue size[1], thereby showing no improvement in the performance with an increase in queue size. To observe the effect of varying input queue size we performed simulations under input load 0.7.

[1]The effect of increasing the buffer size was observed only for small input queue sizes. If the input buffer size is very large or infinite then we may observe an improvement in the performance of the dilated switches under full load

Now the probability of occurrence of a cell at a particular input line is 0.7 rather than 1. The Average number of cells generated in one cycle would be equal to 70% of the size of the switch. Figure 4.5 shows that dilation degree 2 is sufficient to obtain CLP around 10^{-6} for various switches at 0.7 load and input buffer size 5. In Figure 4.6 the improvement in CLP is very clear at input load 0.7 for 32×32. To obtain a CLP around 10^{-6} for a 32×32 switch a dilation degree of 3 is required at input queue size 1. Whereas when the queue size is increased to 9 a dilation degree of 2 is sufficient.

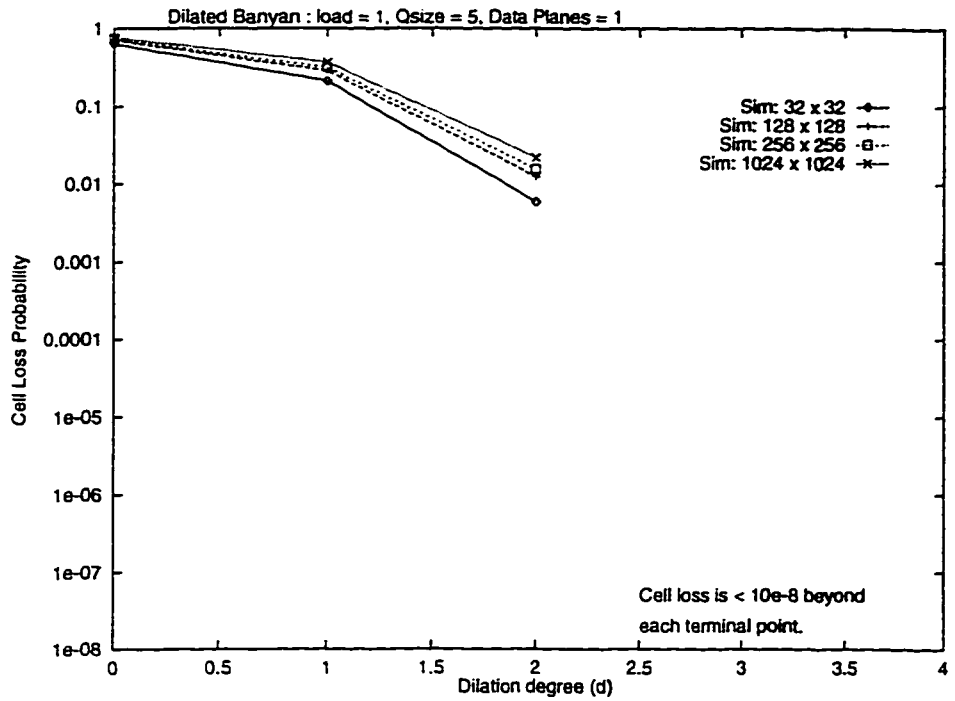

Figure 4.3: Cell loss probability in buffered dilated banyans at full load.

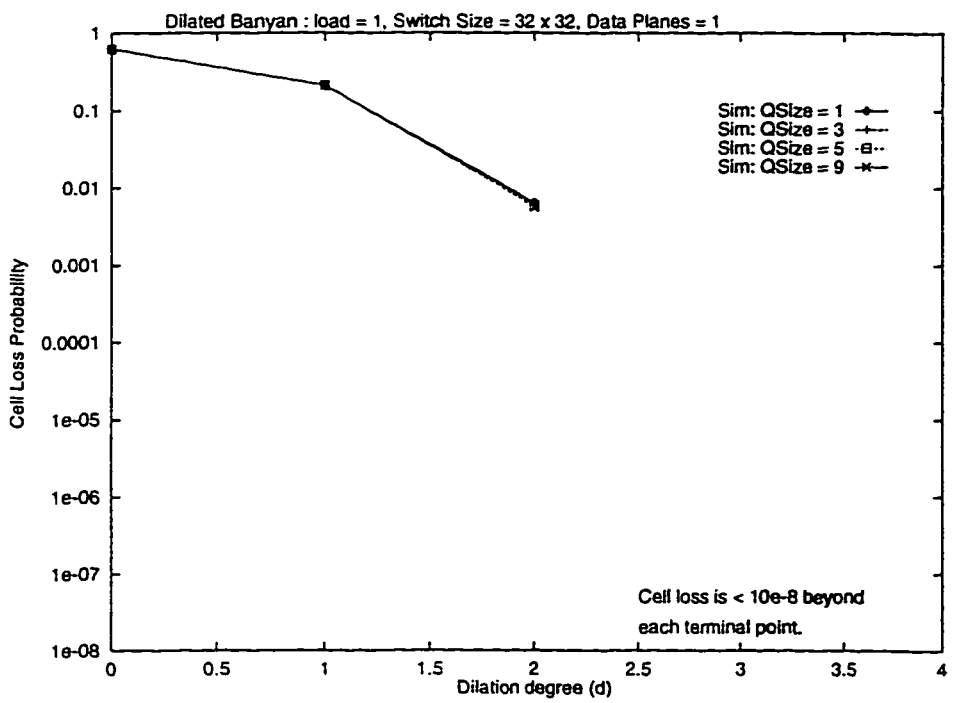

Figure 4.4: Effect of varying input buffer size in a buffered dilated banyan of size 32 x 32 at full load

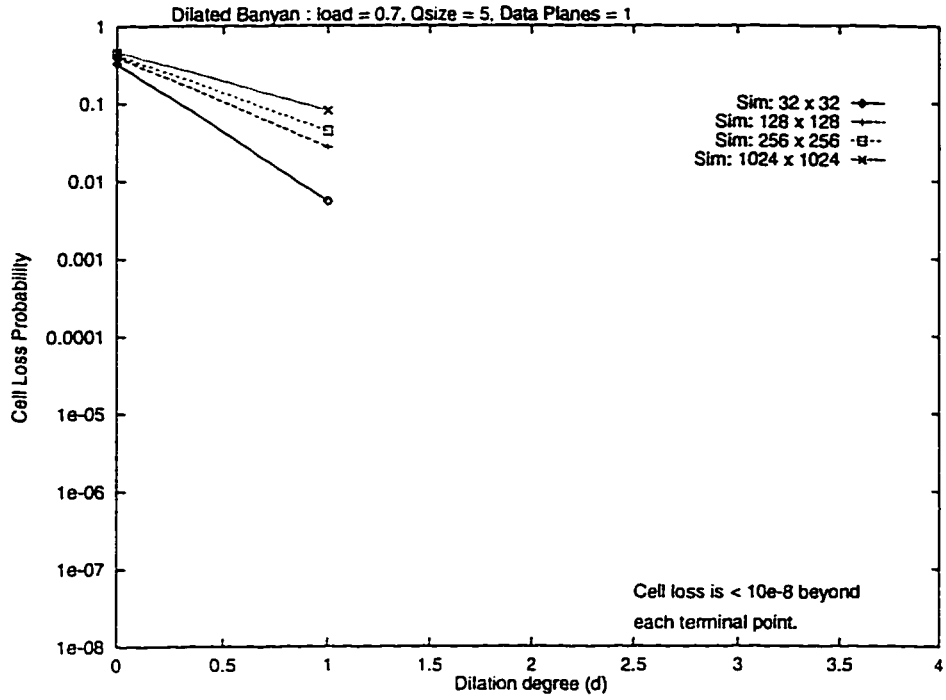

Figure 4.5: Cell loss probability in buffered dilated banyans at partial load.

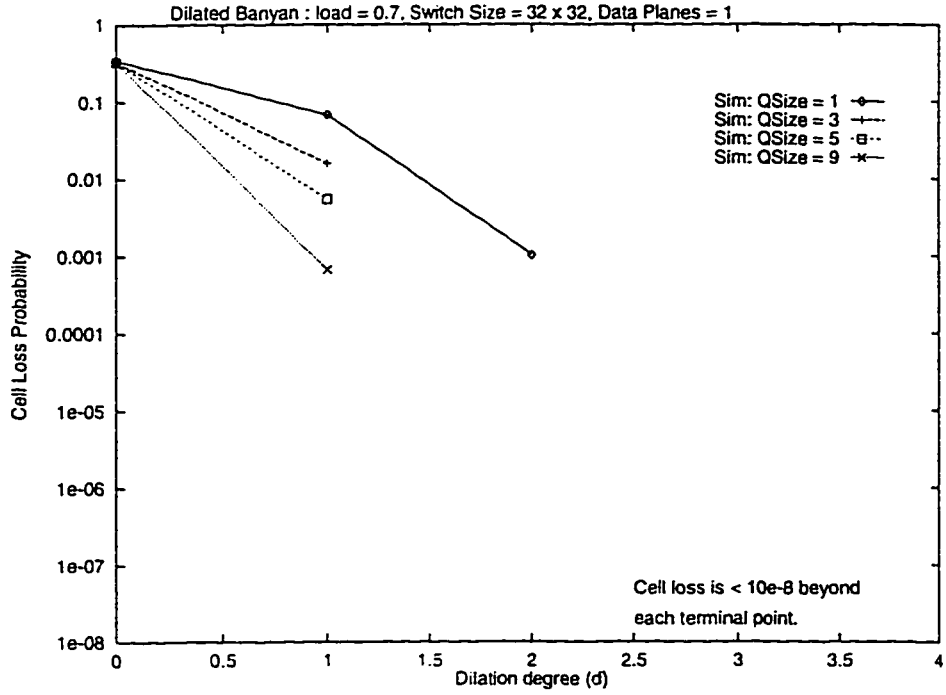

Figure 4.6: Effect of varying input buffer size in a buffered dilated banyan of size 32 x 32 at partial load

4.5 Pipelined Simple Banyan

In this section we will discuss the performance analysis of Pipelined simple banyan proposed by P.C. Wong and M.S. Yeung [25], both by analytical and simulation methods.

4.5.1 Analytical Models

Unbuffered Pipelined Simple Banyan:

In pipelined simple banyan cells are generated at the beginning of a time slot. Each time slot is divided into K reservation slots where K is equal to the number of data planes. Starting from the first data plane cells are routed through each data plane in each reservation slot. Cells which are not successful in reaching their destination due to conflicts are presented to the next data plane. Cells which are not successful in a given time slot i.e. after routing in the last data plane are considered to be lost. If p_0 is the input load at the first data plane, the load at the next data plane would be $p_0 - p'_0$ where p'_0 is the passthrough of the first data plane. Since each data plane is a simple banyan, the passthrough can be obtained by the following recursive formula:

$$p_{j+1} = p_j(1 - p_j/4) \quad for\ 0 \leq j < n \tag{4.3}$$

Solving for p_n in Equation 4.3 at $j = n - 1$ we obtain passthrough of the first data plane as $p'_0 = p_n$. The load at the next data plane would be $p_1 = p_0 - p'_0$. If $p_0, p_1, \ldots, p_{k-1}$ represent the loads at data planes $0, 1, \ldots, k-1$ respectively, and $p'_0, p'_1, \ldots, p'_{k-1}$ are the passthrough of the corresponding data planes, then the cell loss probability of the pipelined simple banyan would be $p_{k-1} - p'_{k-1}$.

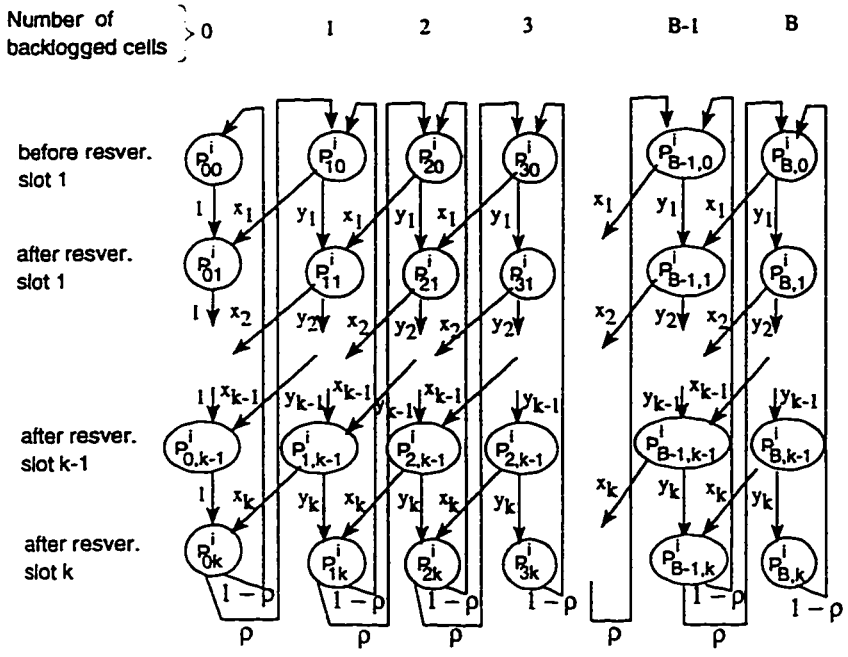

Figure 4.7: State transition diagram for an input queue.

Buffered Pipelined Simple Banyan:

Buffered pipelined simple banyan has queues of a particular size at the input ports to store the incoming cells. An analytical model for buffered pipelined simple banyan has been derived in [25]. The assumptions made in the derivation of this analytical model are that the traffic is uniform and cells arrive with identical and independent probability ρ. The state transition diagram for one single input queue at time slot i is shown in Figure 4.7. The maximum size of the queue is assumed to be B. In the figure, $p^i_{j,k}$ represents the probability of j backlogged cells in the queue at k_{th} reservation slot of the i_{th} time slot. For k_{th} data plane, if its input load is λ_k and its throughput is S_k, then its normalized throughput is given by

$$x_k = S_k/\lambda_k \tag{4.4}$$

x_k represents the probability that a HOL cell is delivered at k_{th} reservation slot and the probability that the cell will stay in the queue is given by $y_k = 1 - x_k$ for $1 \leq k \leq K$ *dataplanes*. The probability state vector at k_{th} reservation slot in i_{th} time slot is given by,

$$\vec{V}_k^i = [p_{0,k}^i \;\; p_{1,k}^i \;\; \cdots \;\; p_{B,k}^i] \tag{4.5}$$

The state transition matrix for k_{th} reservation slot is given by

$$M_k = \begin{pmatrix} 1 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 \\ x_k & y_k & 0 & 0 & \ldots & 0 & 0 & 0 \\ 0 & x_k & y_k & 0 & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \ldots & x_k & y_k & 0 \\ 0 & 0 & 0 & 0 & \ldots & 0 & x_k & y_k \end{pmatrix} \tag{4.6}$$

The state transition matrix for i_{th} time slot is given by

$$A = \begin{pmatrix} 1-\rho & \rho & 0 & 0 & \ldots & 0 & 0 & 0 \\ 0 & 1-\rho & \rho & 0 & \ldots & 0 & 0 & 0 \\ 0 & 0 & 1-\rho & \rho & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \ldots & 0 & 1-\rho & \rho \\ 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 1 \end{pmatrix} \tag{4.7}$$

Starting with some initial vector, say $\vec{V}_0^0 = [1-\rho \;\; \rho \;\; 0 \;\; 0 \;\; \ldots \;\; 0]$, the state vector after K reservation slots can be calculated as,

$$\vec{V}_K^i = \vec{V}_0^i M_1^i M_2^i \ldots M_K^i \tag{4.8}$$

and the state vector for next time slot $i+1$ is calculated as

$$\vec{V}_0^{i+1} = \vec{V}_K^i A \tag{4.9}$$

Applying the above equations 4.9 and 4.8 for a few iterations we can obtain the steady state vector $\vec{V}_k^i = [p_{0,k}^i \;\; p_{1,k}^i \;\; \ldots \;\; p_{B,k}^i]$. The cell loss probability is given by,

$$P_{loss} = \rho \times p_{B,k}^i \tag{4.10}$$

4.5.2 Simulation

We have performed simulations for the pipelined simple banyan under uniform traffic at full load. For unbuffered switch simulations the cells are generated at the beginning of each cycle and applied at the first data plane. Unsuccessful cells in the first data plane are applied to the next data plane. Those cells which are unsuccessful in the last data plane are considered to be lost. Figure 4.8 shows the analytical as well as simulation results. The lines represent the analytical results and the symbols represent the simulation results. For a switch of size 1024×1024 the number of banyans required to achieve a loss of 10^{-6} is 12 which is quite high. This value can be reduced by using input buffers. Figure 4.9 shows the results of buffered pipelined simple banyan with input buffer size equal to 5. Here it can be

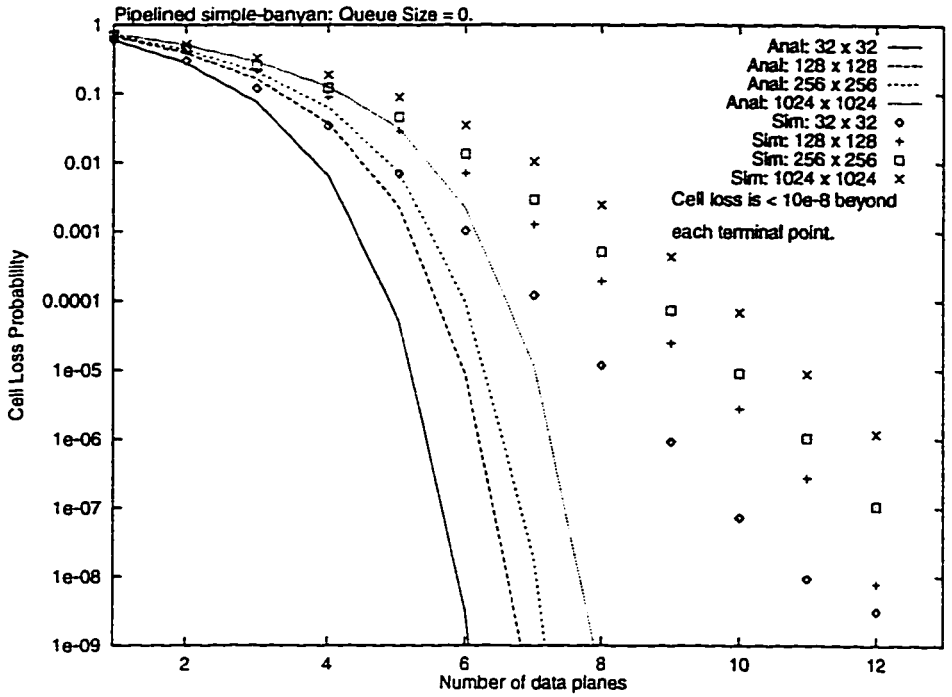

Figure 4.8: Cell loss probability in unbuffered pipelined simple banyans.

seen that for 1024×1024 switch only 6 data planes are required to achieve a cell loss around 10^{-6}. Thus due to input buffering, the number of data planes required to achieve a given cell loss performance has reduced which results in a decrease of not only the hardware resources but also the delay time. The number of data planes can be reduced further by increasing the input queue size. Figure 4.9 shows the effect of increasing input queue size for a 32×32 switch. For a cell loss around 10^{-6} the number of data planes required is 6 when the input queue size 2, while from input queue size of 10 it reduces to 4. Thus, input buffering is useful even at full load in pipelined simple banyan unlike in dilated banyan presented in previous section. In Figures 4.8, 4.9 and 4.10, the lines represent the analytical results and the symbols represent the simulation results. In all these figures the analytical results are more optimistic than simulation results. However the difference between the analytical

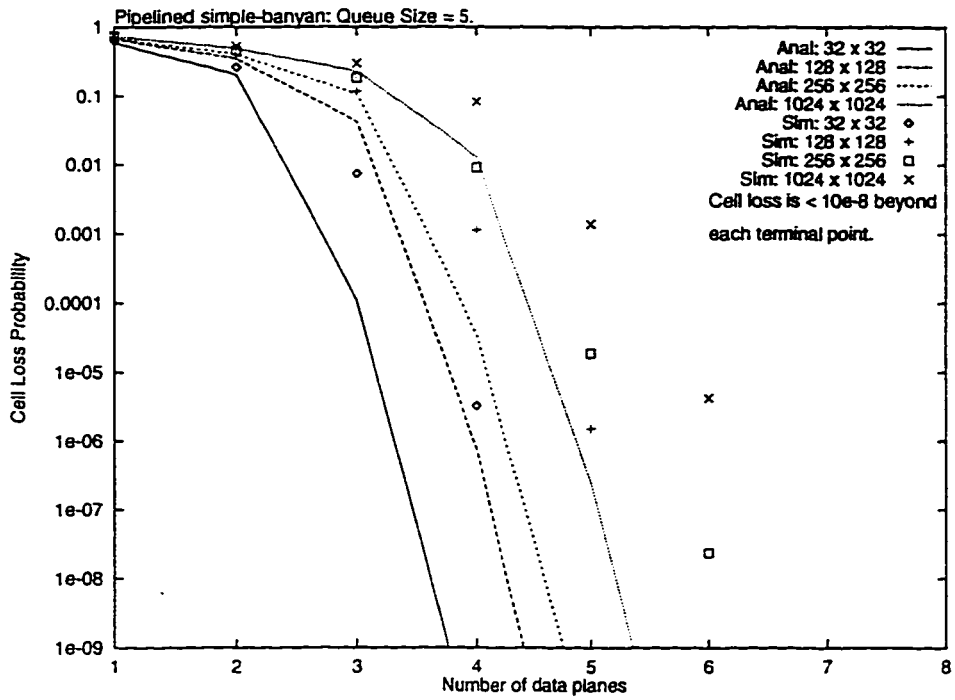

Figure 4.9: Cell loss probability in buffered pipelined simple banyans.

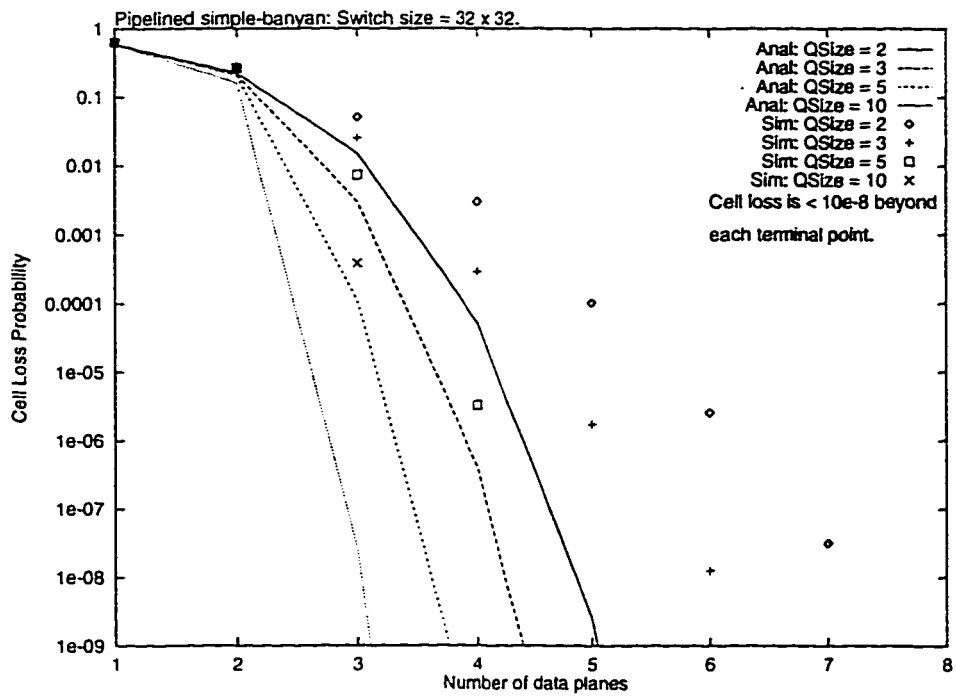

Figure 4.10: Effect of varying input buffer size in a buffered pipelined simple banyan of size 32 x 32

and simulation increases with an increase in the number of data planes which is of order as much as 10^4 for a 1024 x 1024 switch in Figure 4.9. The reason for this difference is that, in the analytical models of the pipelined simple banyans it was assumed that an incoming cell chooses its destination uniformly among all N output ports and independently from all other cell's destination request. That is to say there is no correlation in the output ports request. This assumption is made for all data planes. However, the output ports requested are not correlated only in the first data plane. From the second data plane onwards, the cells become more and more correlated in their output port request. Consider for example that there was a conflict (internal or external) among 4 cells in the first data plane. One cell will be routed while 3 cells will be applied in the next data plane. In the second data plane again there will be conflict among these 3 cells, allowing only one cell to pass. Thus in realistic situation the cells become correlated in their output port destination requests in data planes other than the first data plane. We have simulated the pipelined simple banyan again to observe this effect of output port request correlation. Here, cells were generated in each data plane for all input ports and stored in input queues. The HOL cells were applied to first data plane. Before applying the HOL cells to next data plane their destinations are replaced by another set of randomly regenerated destinations, so that the output ports become uncorrelated as assumed in the analytical model. Figure 4.11, 4.12, 4.13 correspond to 4.8, 4.9, 4.10 respectively with the difference that here the traffic at each data plane is not correlated in its output port request. We can observe that the difference between the analytical and the simulation has reduced significantly in these figures as expected.

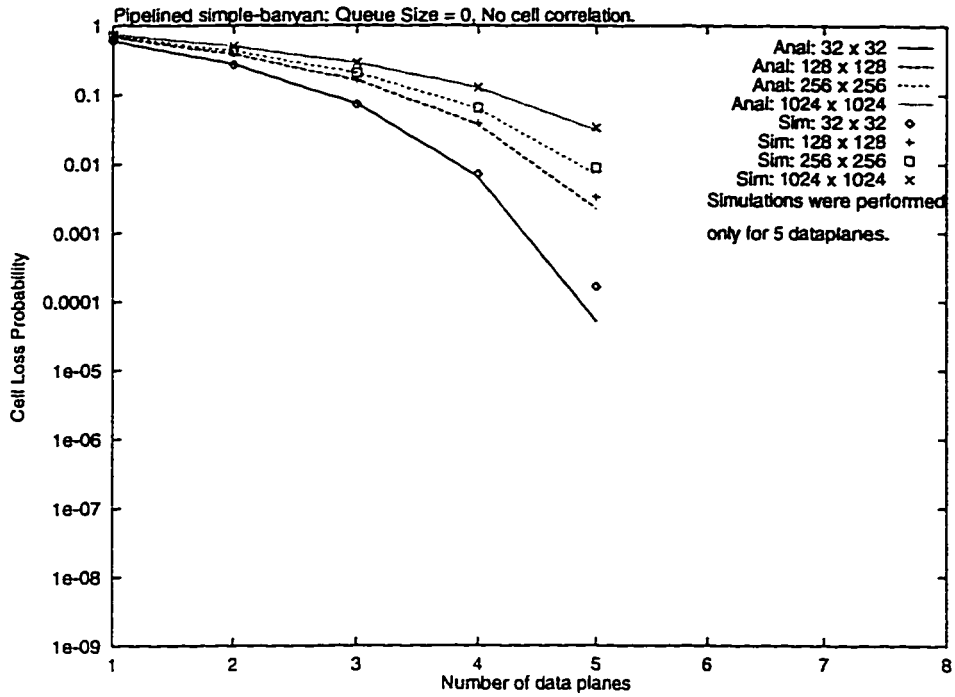

Figure 4.11: Cell loss probability in unbuffered pipelined simple banyans with no output correlation.

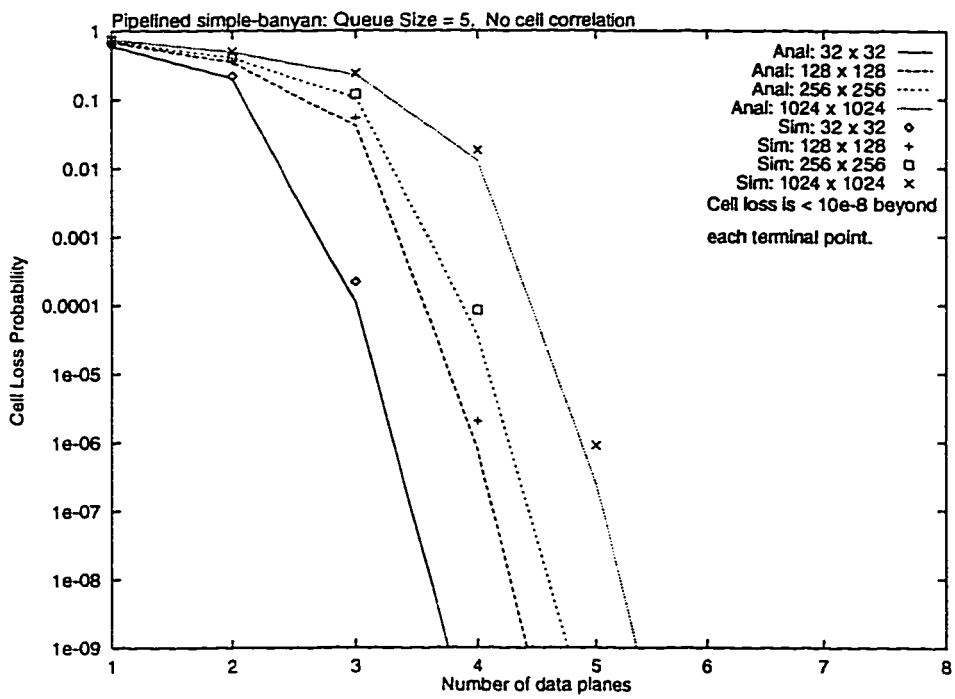

Figure 4.12: Cell loss probability in buffered pipelined simple banyans with no output correlation.

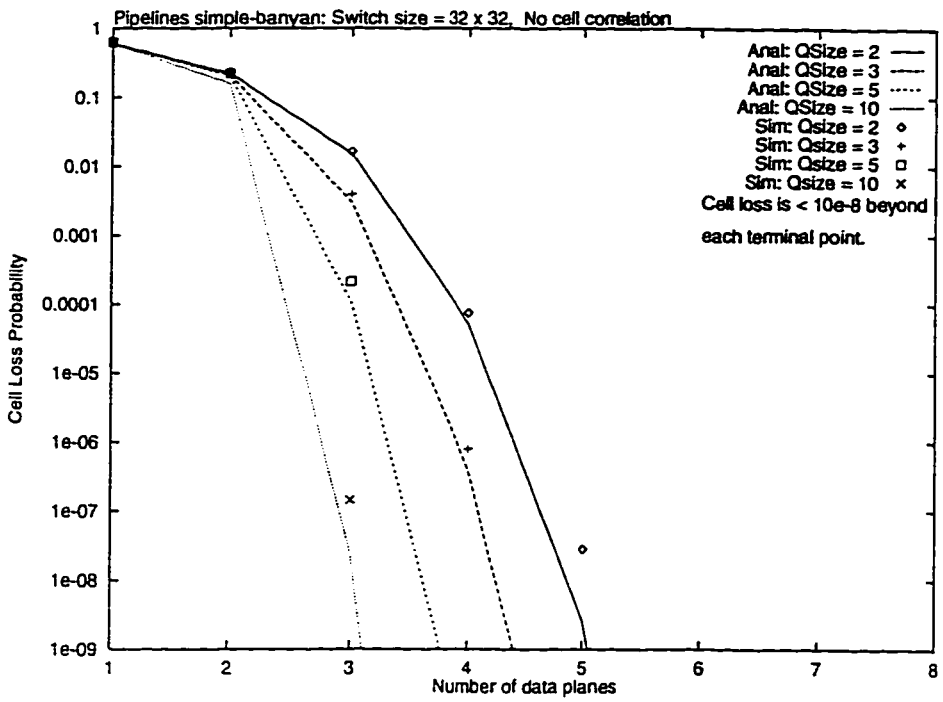

Figure 4.13: Effect of varying input buffer size in a buffered pipelined simple banyan of size 32 x 32 with no output correlation.

4.6 Pipelined Dilated Banyan

Pipelined simple banyan requires a large number of data planes for larger switch sizes, specifically at smaller queue sizes. Queue size is also an important feature of pipelined banyan as it plays an essential role in improving the switch cell loss performance. Larger queue size implies not only an increase in the hardware cost but also an increase in the queuing delay. Therefore it is important to decrease the input queue size to reduce the total switching delay. Using dilated banyan in each data plane instead of simple banyan reduces both the number of data planes and the input queue size. In this section we will discuss the performance analysis of pipelined dilated banyan under uniform traffic and full load.

4.6.1 Analytical Model

Unbuffered Pipelined Dilated Banyan:

A pipelined dilated banyan has dilated banyans in each data plane. The analytical model to calculate the throughput of a dilated banyan was presented in section 4.4.1. Assume that input traffic at the first data plane(D_0) is p_0 and that there are m data planes in the pipelined dilated banyan. The throughput of a data plane can be evaluated by using the equations 4.2 and 4.2. Let the load at daptaplanes $D_0, D_2, \cdots, D_{m-1}$ be represented as $p_0, p_1, \cdots, p_{m-1}$ and the passthrough as $p'_0, p'_1, \cdots, p'_{m-1}$ respectively.

The Load at Dataplane $i+1 = p_{i+1} = p_i - p'_i$, for $0 \leq i \leq m-2$

where p_i is load at data plane i and p'_i is the throughput of data plane i.

Cell loss probability of the dilated pipelined banyan $= p_{m-1} - p'_{m-1}$.

Buffered Pipelined Dilated Banyan:

The architecture of buffered pipelined dilated banyan is similar to buffered pipelined banyan with the exception that each data plane here has a dilated banyan instead of simple banyan. Therefore the state diagram of a single input queue in a buffered dilated banyan would be the same as shown in Figure 4.7. The probability state vector $\vec{V}^i$, state transition matrix over each reservation slot M, and the state transition matrix over each time slot A would also be the same as shown in equations 4.5, 4.6 and 4.7 respectively. However the throughput S_k in equation 4.4 for each data plane would be evaluated by the method shown above. The cell loss probability of buffered dilated banyan can be evaluated $p_{loss} = \rho \times P_B^i$ where ρ is the input traffic load and P_B^i is probability that the input queue is full.

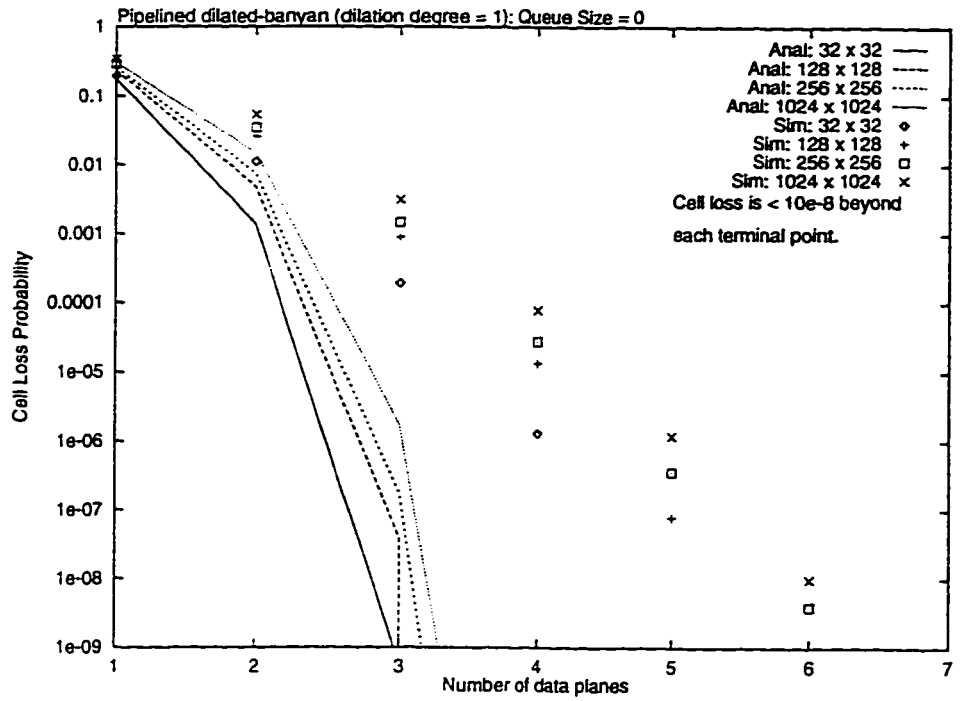

Figure 4.14: Cell loss probability in unbuffered pipelined dilated banyans.

4.6.2 Simulation

We have simulated the buffered pipelined dilated banyans under uniform traffic, full load and dilation degree 1 i.e. dilated banyan in each data plane has 2 links per output port. The simulations were performed for switch sizes $32 \times 32, 128 \times 128, 256 \times 256$ *and* 1024×1024, at various input buffer sizes like 0, 2, 3, 5 and 10. Figure 4.14 shows the analytical and simulation results for various pipelined dilated banyans at input queue size 0. The simulation results predict that a CLP of 10^{-6} can be achieved with only 5 data planes for 1024×1024 as compared to 12 required in pipelined simple banyan (Figure 4.8). This is because the throughput of each data plane. in pipelined dilated banyan is much higher than in pipelined simple banyan. It was shown in pipelined simple banyan that with input buffering, the performance of pipelined banyan increases. Figure 4.15 shows the results of

buffered pipelined dilated banyan for various switches at input queue size 5. With buffering, the number of data planes reduces from 5 in unbuffered pipelined dilated banyan to 3. The number of data planes required was 6 in pipelined simple banyan for a similar situation. Figure 4.16 shows the effect of increasing the input queue size on a 32 x 32 switch. At queue size 2 the CLP for 32 x 32 switch is 10^{-3} which reduces to 10^{-7} at queue size 5 and less than 10^{-9} for queue size 10.

In Figures 4.14, 4.15 and 4.16 we can see that the difference between the analytical and simulation results is very large. This is due to the traffic getting correlated in successive data planes in pipelined banyans as explained in pipelined simple banyan section. We have simulated the unbuffered and buffered pipelined dilated banyans so that the traffic is uncorrelated in destination port request in each data plane. Figures 4.17, 4.18 and 4.19 correspond to Figures 4.14, 4.15 and 4.16 respectively with a difference that here simulations were performed by randomizing the cell distribution at each data plane. It can be seen that the difference between the analytical and simulation results has decreased here. However this difference is not as close as it was observed for the pipelined simple banyan case under the same conditions. For example the error difference in the analytical and simulation for a switch of size 1024 x 1024 in Figure 4.17 is 10^{-1}. This relatively large difference can be attributed to the assumptions made in the analytical modeling of the D-SW switch in the dilated banyan. It was assumed that the probability of a cell arrival is the same at all the inputs of the D-SW switching elements at each stage in the router part of the dilated banyan. The cells become more and more correlated at each stage in the dilated banyan in the simulation. This may lead to an increase in the cell loss, thus making the difference with the analytical results.

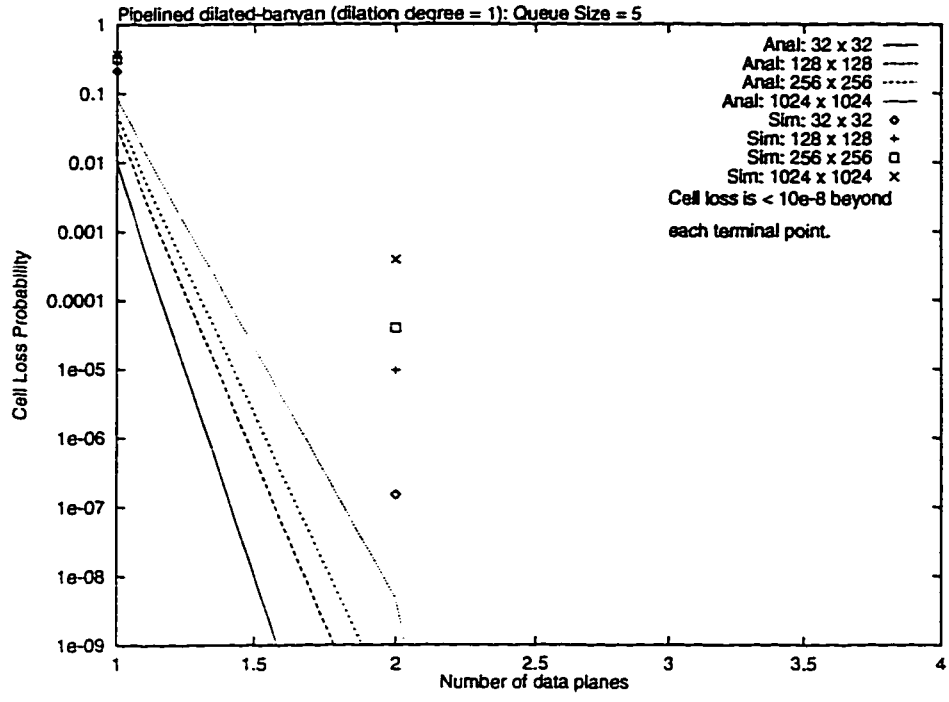

Figure 4.15: Cell loss probability in buffered pipelined dilated banyans.

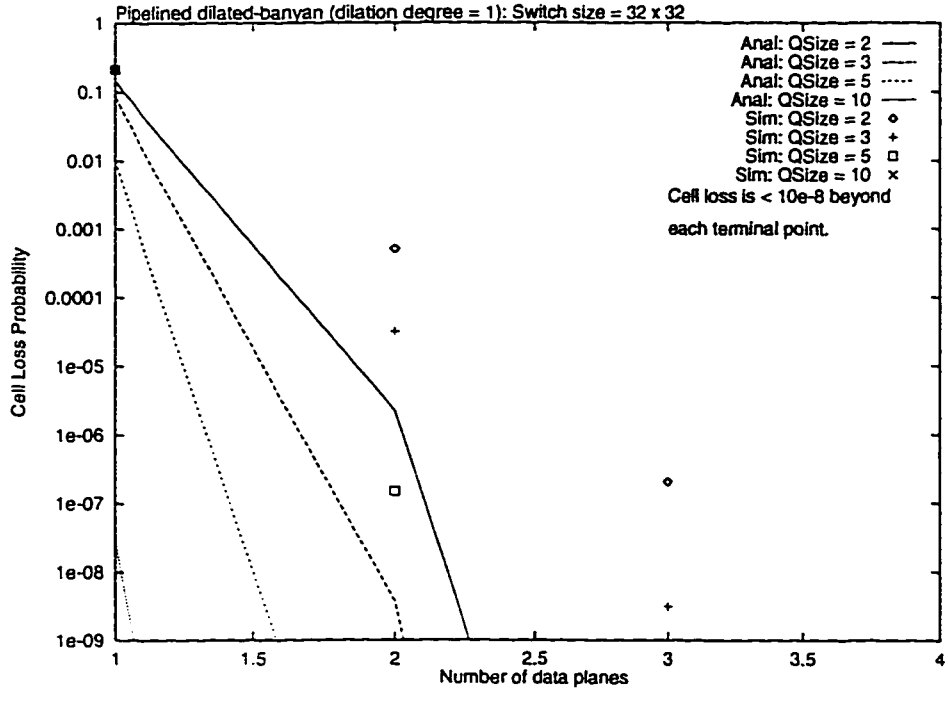

Figure 4.16: Effect of varying input buffer size in a buffered pipelined dilated banyan of size 32 x 32

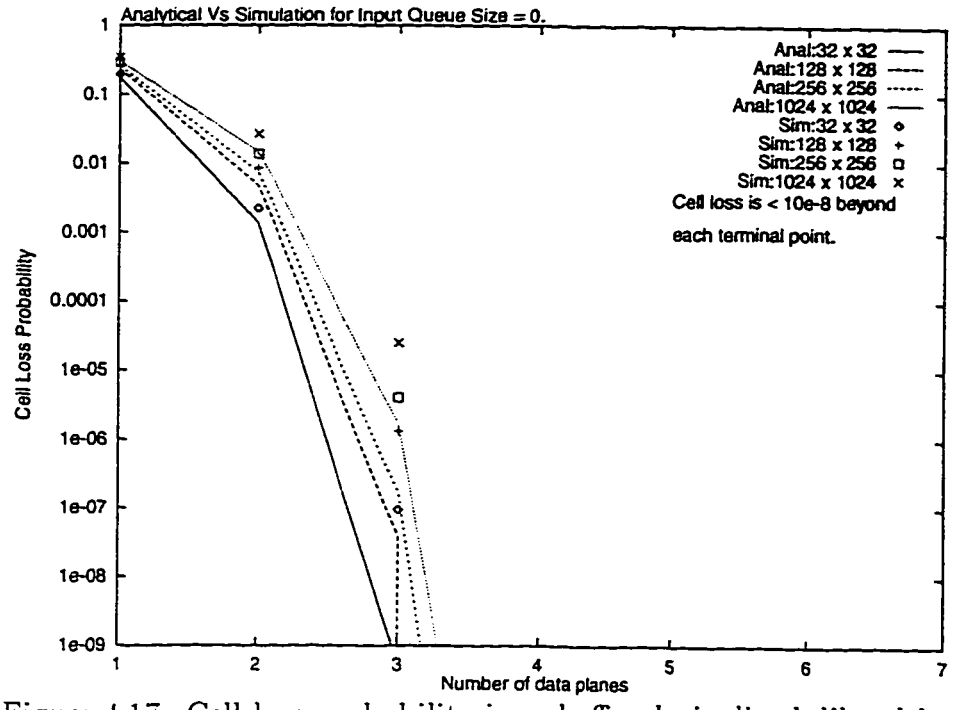

Figure 4.17: Cell loss probability in unbuffered pipelined dilated banyans with no output correlation.

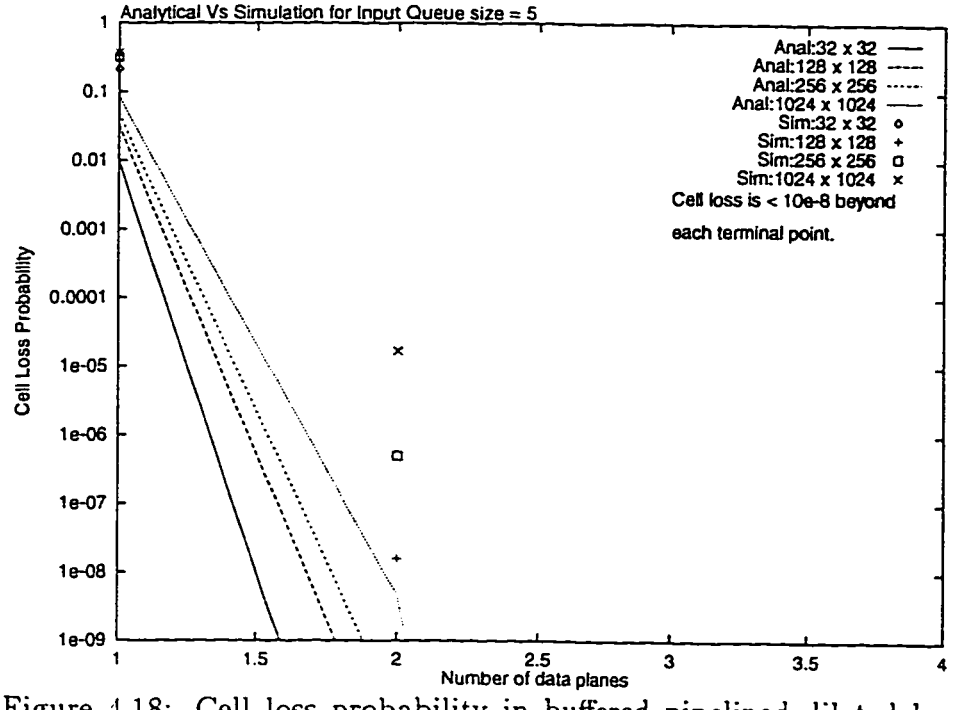

Figure 4.18: Cell loss probability in buffered pipelined dilated banyans with no output correlation.

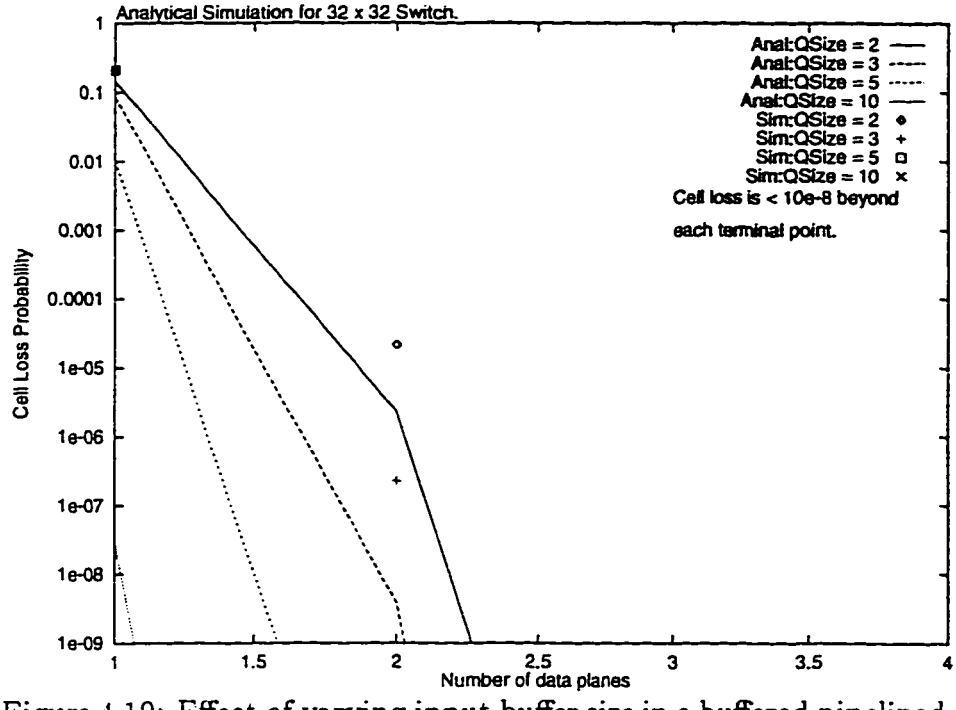

Figure 4.19: Effect of varying input buffer size in a buffered pipelined dilated banyan of size 32 x 32 with no output correlation.

4.7 Pipelined Expanded Banyan

In the previous section, we found that pipelined dilated banyan has reduced the input queue size and the number of data planes required as compared to pipelined simple banyan to achieve a given cell loss probability. However it was shown in section 3.3.1 that the complexity of a dilated banyan grows very fast with an increase in dilation degree (grows as 2^d). Therefore a pipelined dilated banyan which uses dilated banyan in each data plane has higher complexity as compared to pipelined simple banyan even if it uses very small dilation factor. An Expanded switch fabric (EBSF) has lesser complexity than the dilated banyan. The delay in EBSF is always the same for a particular switch size with any amount of expansion factor, whereas in dilated banyan the delay would increase with an increase in dilation factor for a particular switch size. Therefore we have simulated pipelined expanded banyan

which has expanded banyan in each data plane rather than dilated or simple banyan. In this section we will discuss the performance of a pipelined expanded banyan.

Pipelined Expanded Banyan with expansion factor 2:

An expanded banyan switch fabric with expansion factor EF has EF banyans interleaved. There are EF output links per output port. The first EF stages in this expanded switch has only demultiplexers. An EBSF of size 8×8 with $EF = 4$ is shown in Figure 2.8 in section 2.1.4. In this figure a conflict free path is guaranteed in first two stages. Figure 4.20 presents the simulation results of buffered pipelined expanded banyan switches with expansion factor 2 and input queue size 5. For a switch of any size, 4 data planes are enough to obtain a CLP less than 10^{-6} and with 5 data planes we can achieve a CLP around 10−9. Figure 4.21 illustrates the effect of increasing the input buffer size on a 32×32 input buffered pipelined expanded banyan with expansion factor 2. The number of data planes reduces from 5 to 3 when the input queue size is increased from 2 to 10.

Pipelined Expanded Banyan with expansion factor 4:

For a given performance measure, the number of data planes required in pipelined expanded banyan with expansion factor 2 is less than that required in a pipelined simple banyan but is greater than in pipelined dilated banyan. This is because the throughput of a data plane in pipelined expanded banyan with expansion factor 2 lies between the throughput of a simple banyan and a dilated banyan with dilation factor 1. We can increase the expansion factor of each data plane to improve the throughput of each data plane without effecting the delay. Therefore to further reduce the number of data planes required we performed simulations for pipelined expanded banyan with expansion factor 4. Figure 4.23 shows that the number of

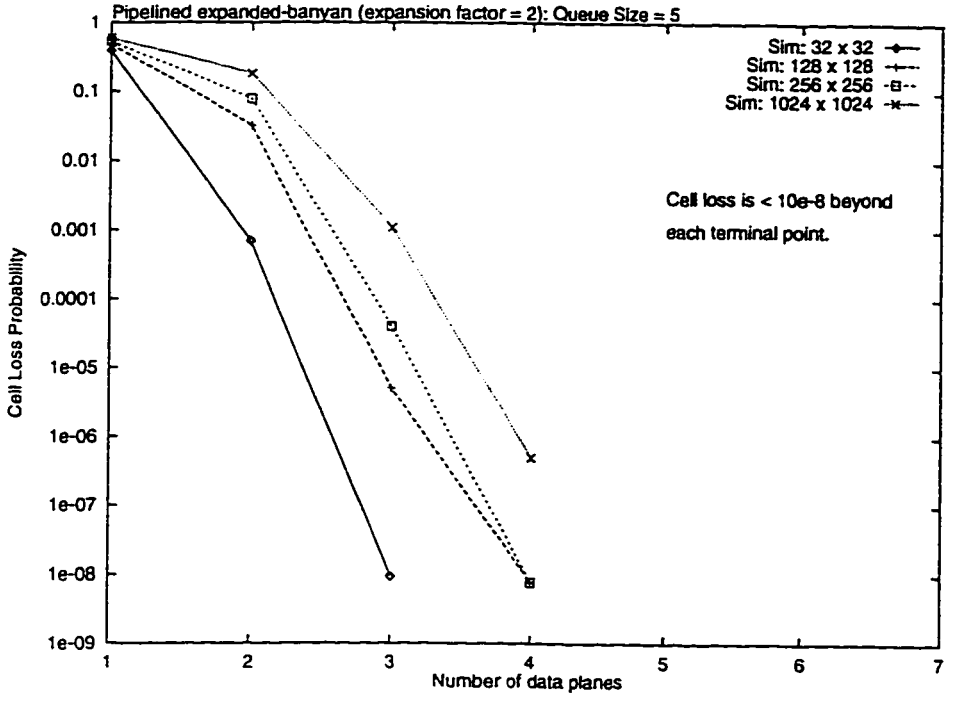

Figure 4.20: Cell loss probability in buffered pipelined expanded banyans with expansion factor 2.

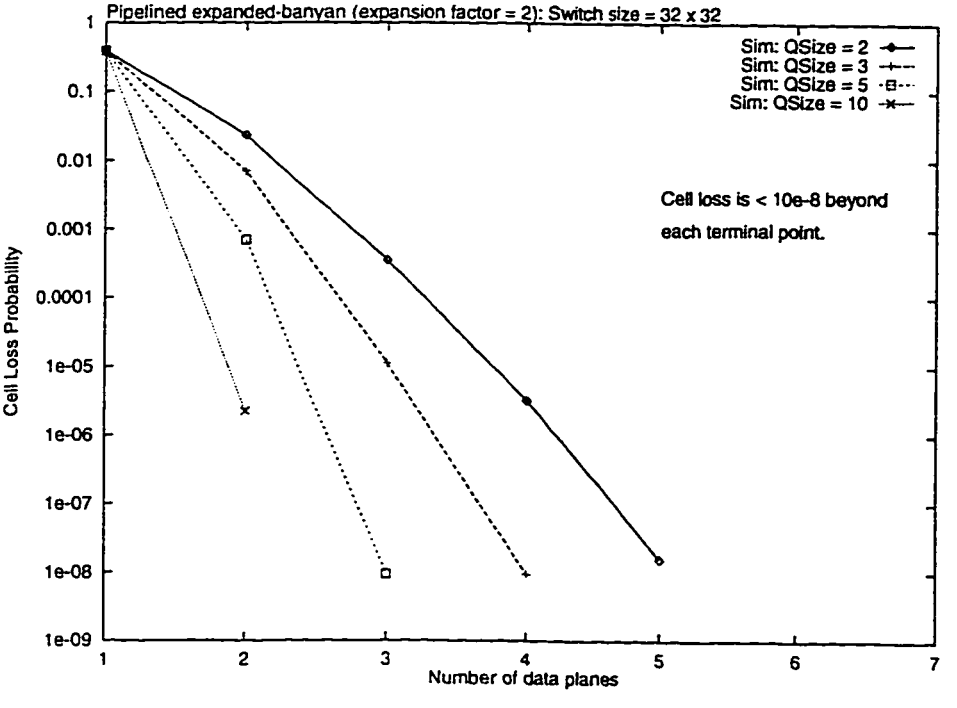

Figure 4.21: Effect of varying input buffer size in a buffered pipelined expanded banyan of size 32 x 32 with expansion factor 2.

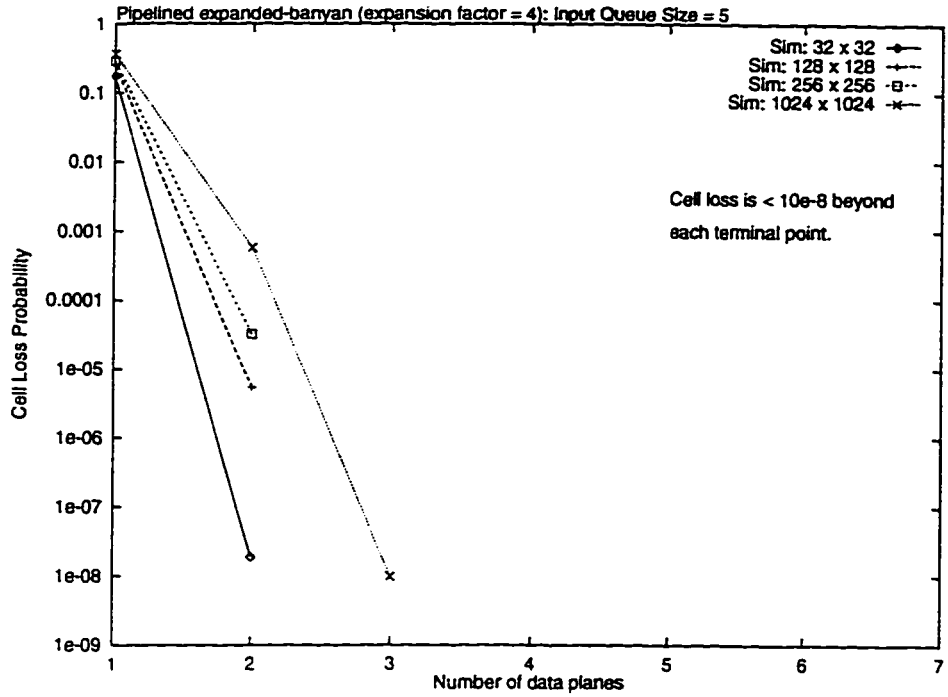

Figure 4.22: Cell loss probability in buffered pipelined expanded banyans with expansion factor 4.

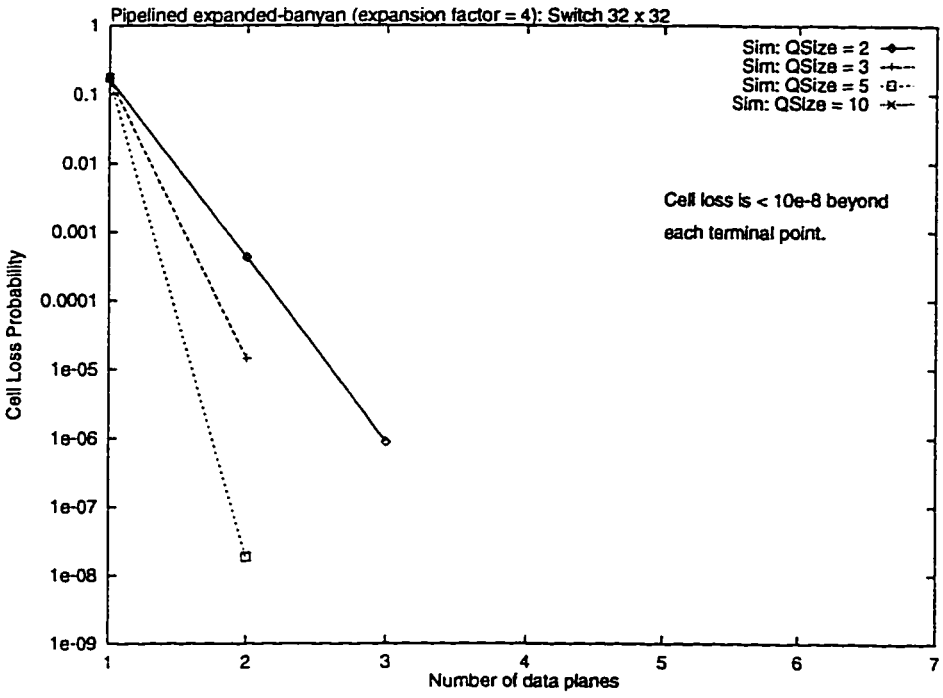

Figure 4.23: Effect of varying input buffer size in a buffered pipelined expanded banyan of size 32 x 32 with expansion factor 4.

data planes reduces from 5 in a switch with expansion factor 2 to 3 in a switch with expansion factor 4. Figure 4.23 illustrates the effect of increasing the input queue size on a 32 × 32 pipelined expanded banyan switch for expansion factor 4.

4.8 Comparison

In previous sections we have presented the results of four different pipelined switches which are: pipelined simple banyan (PSB), pipelined dilated banyan with dilation degree 1 (PDB_2), pipelined expanded banyan with expansion factor 2 (PEBSF_2) and pipelined expanded banyan with expansion factor 4 (PEBSF_4). In this section we present a comparison of these four different types of switches for the largest switch size 1024 × 1024 under uniform traffic and full load. Figure 4.24 shows the simulation results of these four types of switches for input queue size 2. It can be seen that at small input queue sizes the performance of PSB is low while PEBSF_2 performs better. The performance of PEBSF_4 is close to PDB_2 for higher cell loss probabilities. The difference between them increases for lower cell loss probability. Figure 4.25 shows the simulation results of a 1024 × 1024 switch for various types of pipelined banyans with input queue size 10. The difference between PSB and PDB_2 is still the same, but the PEBSF_10 performs similar to PDB_2 with its curve overlapping that of the PDB_2 in Figure 4.25.

4.9 Conclusion

In this chapter we have presented step-by-step the results of dilated banyan, pipelined simple banyan, pipelined dilated banyan with dilation degree 1, pipelined expanded

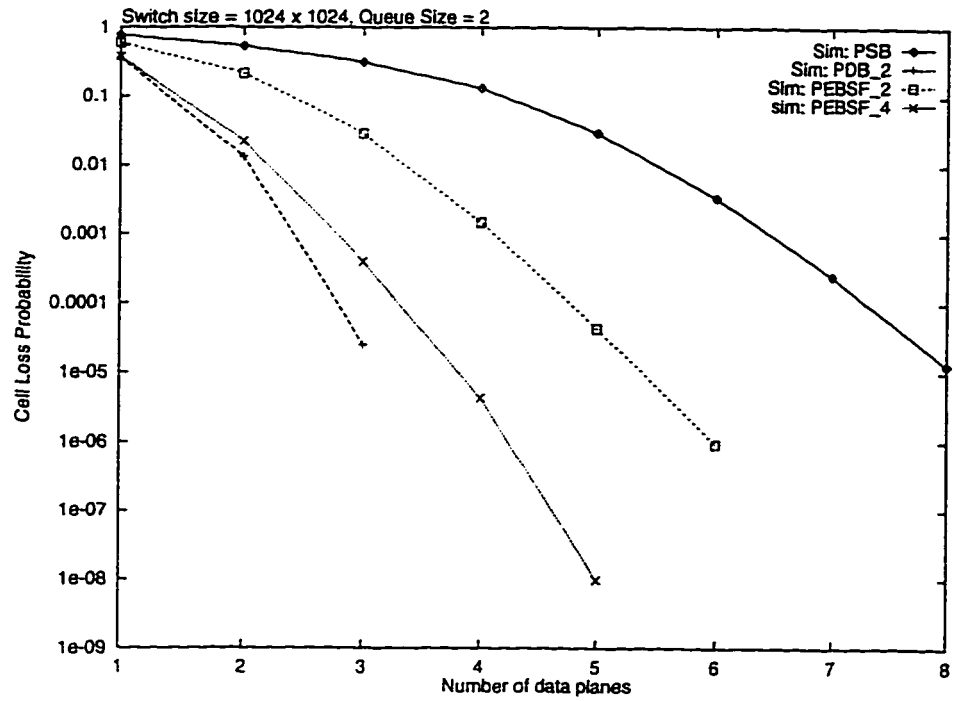

Figure 4.24: Comparison of buffered pipelined simple banyan, pipelined dilated banyan and pipelined expanded banyan with input queue size 2.

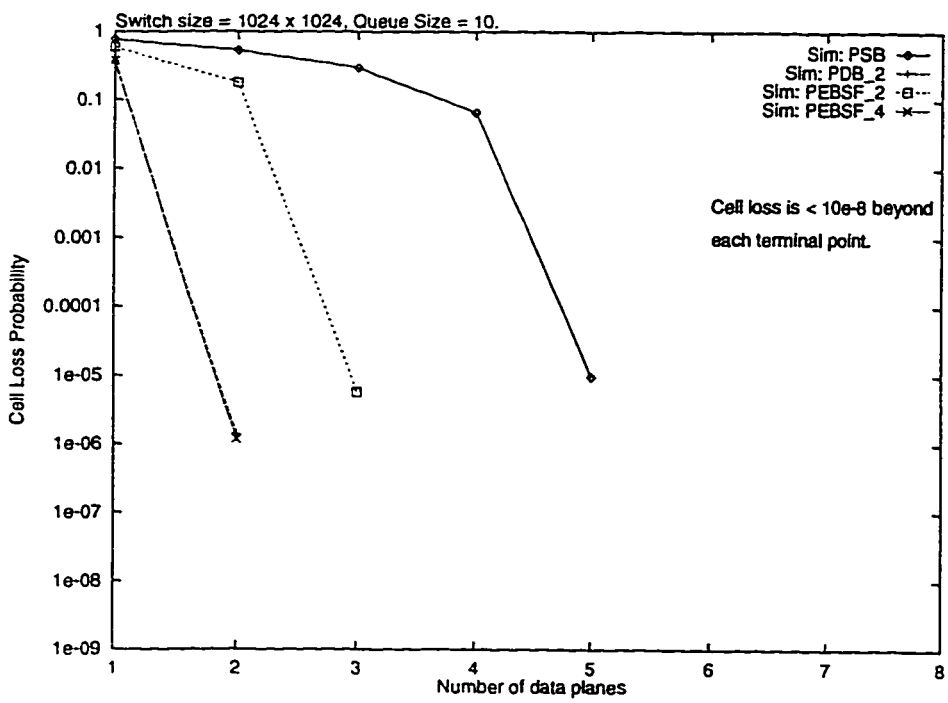

Figure 4.25: Comparison of buffered pipelined simple banyan, pipelined dilated banyan and pipelined expanded banyan with input queue size 2.

banyan with expansion factor 2 and pipelined expanded banyan 4. All the above simulations were performed under uniform traffic and full load [2]. The results were shown for switches 32×32, 128×128, 256×256 and 1024×1024 with various input queue sizes of 0,2,3,5 and 10. The pipelined simple banyans, although required less hardware, the performance was low specifically for large switch sizes. Pipelined dilated banyan showed a large improvement in performance with a dilation degree of only 1. However the complexity for this was found to be very high due to the presence of dilated banyans in each data plane. Pipelined expanded banyan with expansion factor 2 showed better results than pipelined simple banyan but its performance was not as good as pipelined dilated banyan. Finally a pipelined expanded with expansion factor 4 and input queue size 10 gave performance similar to a pipelined dilated banyan. Pipelined expanded banyan with expansion factor 4 is better than pipelined dilated banyan because the switching delay would be $O(logN)$ for any value of expansion factor. In the next chapter we will present the performance of these different types of pipelined banyans under simulated ATM traffic.

[2]With an exception in case of dilated banyan where the simulations were performed for load 0.7 also.

Chapter 5

Performance Analysis under ATM Traffic

5.1 Introduction

ATM is the proposed transfer mode for BISDN and is expected to support virtually all existing and emerging communication services. The ATM Forum has specified five "Service categories" in relation to traffic management in ATM networks.

Constant Bit Rate(CBR): CBR traffic includes any source which has a continuous stream of bits at a predefined constant rate. A few examples of this type of traffic sources are voice, circuit emulation, some type of video. The requirements for this type of traffic are short delay and low delay jitter.

Real time Variable Bit Rate (rt-VBR): The data rate of the traffic source will be varying here. Compressed video and compressed voice are some examples of this traffic source. This has low transit delay, low delay jitter and very low cell loss requirements.

Non-Real-Time Variable Bit Rate(nrt-VBR): This is for the applications with less restrictions on transit delay and jitter requirements. An example here might be MPEG-2 encoded video distributions. The cell loss should be very low since the loss of a cell in compressed video has a severe effect on the quality of the connection.

Unspecified Bit Rate(UBR): The UBR service is a "best effort" delivery service. This is a connection-less service and the packets may be dropped if there is a congestion on the network. In cases with appropriate end-to-end error recovery protocols this is an acceptable service.

Available Bit Rate(ABR): The concept of ABR is to offer a guaranteed delivery service with minimal cell loss to users who can tolerate a large variation in throughput rate and transit delay. The idea is to evaluate dynamically the available bandwidth on a running network and use it for applications with vague requirements on throughput and delay. The ABR service can change the bit rate of the connection dynamically as the network condition changes either by providing feedback from the network to the sender or by monitoring the network's behavior. Examples of applications that may use this service are file transfer, e-mail, LAN Emulation, etc. For all the above services except for UBR, a source must specify its Quality of Service(QoS) requirements at the time of connection establishment. Some of the QoS parameters are Peak Cell Rate(PCR), Sustainable (average) Cell Rate(SCR), Minimum Cell Rate(MCR) and Maximum Burst Size(MBS). Applications with CBR traffic may have to specify only PCR, while those with rt-VBR traffic will have to specify both PCR and SCR. Sources with nrt-VBR must specify PCR, SCR and MBS. Finally, ABR traffic sources must declare PCR and its MCR.

5.2 ATM Traffic Source Modelling

Traffic Source modelling is the process of capturing the characteristics of a traffic source to define precisely its behavior. It is quite useful in network management by providing several services. For example it could be used in negotiation of Q0S, congestion control, management of connection acceptance for various services etc. A traffic source model is also useful in simulators to evaluate the cell generation times for various types of traffic sources.

Several models have been proposed to describe the behavior of various types of traffic sources. Some of the general source models are Generally Modulated Deterministic Process (GMDP), Markov Modulated Poisson Process (MMPP) and ON/OFF source model. According to GDMP, the source can be in one of N possible states. In a particular state cells are generated at a constant rate. The time spent in a state is geometrically distributed and the next state is determined from a transition probability matrix. The disadvantage of this model is that it requires a number of states and parameters and it is difficult to obtain an unbiased estimation of various parameters. MMPP model is a doubly stochastic Poisson process with a continuous-time m-state Markov chain. The sojourn time for each state is exponentially distributed, and in a particular state cells are generated according to Poisson process.

ON/OFF source model is one of the simplest and most widely used model. This model assumes that the source alternates between *active (busy)* period and *idle (silence)* period as shown in Figure 5.1. In active period the source is in ON state and is transmitting cells at a give rate and in OFF state the source is idle. Different active and idle periods are assumed to be independent from each. Each of these active and idle periods can be either exponential or geometric random variable depending

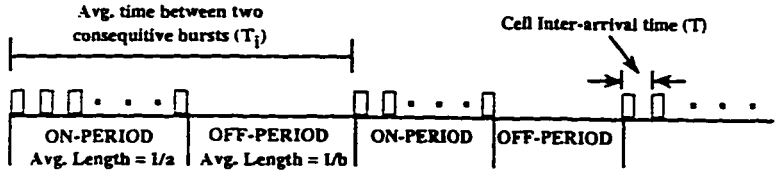

Figure 5.1: On-Off source model.

upon the time axis as either continuous or slotted. The following parameters are used to characterize a particular traffic source in ON/OFF source model

- m : Sustainable Cell Rate (SCR) is the average cell arrival rate.
- B : Mean burst length defines the average number of cells in a burst.
- β : Burstiness is defined as the ratio between the peak cell rate and average cell rate.

The above three parameters are sufficient to model any type of VBR traffic sources using the ON/OFF source model. For CBR traffic only the peak cell rate p is enough since the cells are generated continuously at a constant pace. The minimum interarrival time between two consecutive cells would be reciprocal of the peak arrival rate p i.e. $T = 1/p$. For VBR sources all the three parameters must be specified. Given the average cell arrival rate m, mean burst length B and burstiness factor β the mean duration of *active* and *idle* periods are evaluated as follows:
Peak cell arrival rate $= p = m \times \beta$.
Minimum inter-arrival time between two cells in an active period = T

$$\mathrm{T} = 1/p \tag{5.1}$$

Type of Source	B in cells	Average bit rate $m \times 384$ *bps*	Burstiness β	Cell Loss Tolerance
CBR	N/A	64 Kbps	1	10^{-4} to 10^{-6}
Connectionless data	200	700 Kbps	as high as 1000	10^{-12}
Connection oriented data	200	25 Mbps	as high as 1000	10^{-12}
VBR video	2	25 Mbps	2 to 5	10^{-10}
Background data/video	3	1 Mbps	2 to 5	10^{-9} to 10^{-10}
VBR video/data	30	21 Mbps	2 to 5	10^{-9}
Slow video	3	6 Mbps	2 to 5	10^{-12}

Table 5.1: Parameter values for typical VBR traffic sources.

$$\text{Mean value of active period} = \mathrm{a}^{-1} = B \times T \tag{5.2}$$

Average cell arrival rate $= m = p \times \frac{a^{-1}}{a^{-1}+b^{-1}}$

$$\text{Mean value of Idle Period} = \mathrm{b}^{-1} = \frac{p}{m}(\beta - 1) \quad since\ \beta = p/m \tag{5.3}$$

Table 5.1 shows the parameters for typical VBR traffic sources, as proposed by CCITT (now known as ITU) [28]. They can be used in ON/OFF source models to model a particular VBR traffic source. In this table the inter-burst and burst length are assumed to be exponentially distributed.

5.3 Simulations

In this section, we present the cell loss characteristics of the pipelined simple banyan, pipelined dilated banyan and pipelined expanded banyan switches when subjected to the following two types of traffic mixes:

Traffic Mix 1: CBR: $p = 0.064$ Mbps, #Channels = 10%.

CBR: $p = 1.4$ Mbps, #Channels = 10%.

VBR: $m = 0.7$ Mbps, $B = 200$, $\beta = 5$, #Channels = 20%.

VBR: $m = 25$ Mbps, $B = 20$, $\beta = 5$, #Channels = 20%.

VBR: $m = 21$ Mbps, $B = 30$, $\beta = 4$, #Channels = 40%.

Traffic Mix 2: CBR: $p = 0.064$ Mbps, #Channels = 25%.

CBR: $p = 1.4$ Mbps, #Channels = 25%.

VBR: $m = 0.7$ Mbps, $B = 200$, $\beta = 5$, #Channels = 12%.

VBR: $m = 20$ Mbps, $B = 25$, $\beta = 5$, #Channels = 13%.

VBR: $m = 2$ Mbps, $B = 25$, $\beta = 10$, #Channels = 6%.

VBR: $m = 3$ Mbps, $B = 1$, $\beta = 5$, #Channels = 6%.

VBR: $m = 30$ Mbps, $B = 21$, $\beta = 4$, #Channels = 6%.

VBR: $m = 3$ Mbps, $B = 6$, $\beta = 5$, #Channels = 7%.

We have developed a simulator which can generate any combination of ATM traffic shown in 5.1. The simulator uses the ON/OFF source traffic model. For VBR traffic sources four input parameters m, B, β and the percentage of channels with this type of traffic source are required, while for CBR sources only p and percentage of channels are required. For VBR sources the value of p is calculated from m and β. Then for each type of traffic source the value of T is calculated. The traffic at various sources is synchronized with the source having minimum value of T.

5.3.1 Performance under Traffic Mix 1.

In this section we present the simulations of buffered pipelined switches under ATM traffic mix-1 described above. Here we present two types of results for each of the four types of switches, pipelined simple banyan, pipelined dilated banyan, pipelined expanded banyan with expansion factor 2, and pipelined expanded banyan with expansion factor 4. For a particular banyan, the first plot shows the performance of different switch sizes under a constant input queue size(5) and the second plot shows the influence of varying input queue size on a particular switch (1024×1024). Figures 5.2 and 5.3 present the results of pipelined simple banyan. When these results are compared with the uniform traffic results of previous chapter we observe that the number of data planes required here is less. This is because the load under ATM traffic is less than full load. Figures 5.4 and 5.5 show the results of pipelined dilated banyan with dilation degree 1. The number of data planes required in pipelined dilated banyan is found to be half that of pipelined simple banyan. The results of pipelined expanded banyan with expansion factor 2 are shown in Figures 5.6 and 5.7. The performance of this switch lies between pipelined simple banyan and pipelined dilated banyan. Finally in Figures 5.8 5.9 we present the results of pipelined expanded banyan with expansion factor 4. Results of this switch are similar to those of pipelined dilated banyan.

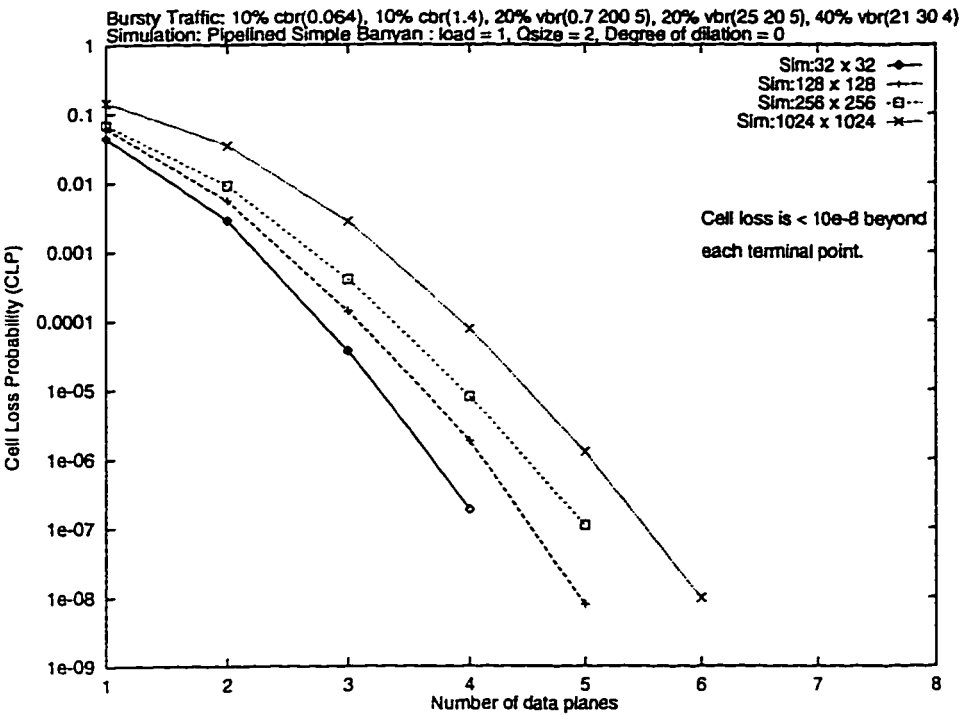

Figure 5.2: Performance of pipelined simple banyans under ATM traffic mix-1.

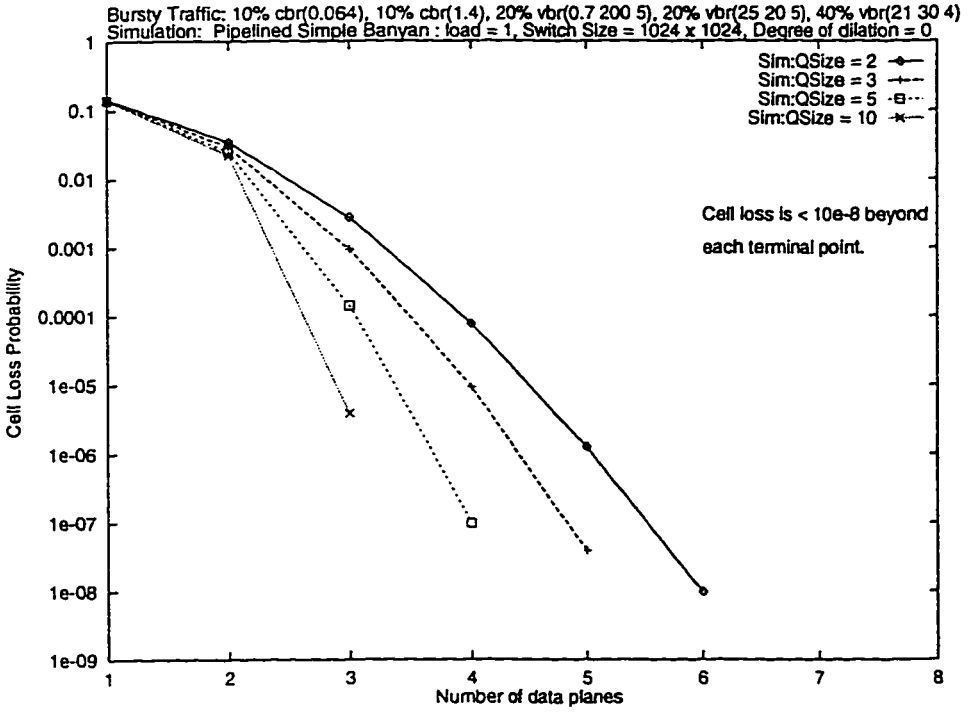

Figure 5.3: Effect of varying buffer size on pipelined simple banyan under ATM traffic mix-1.

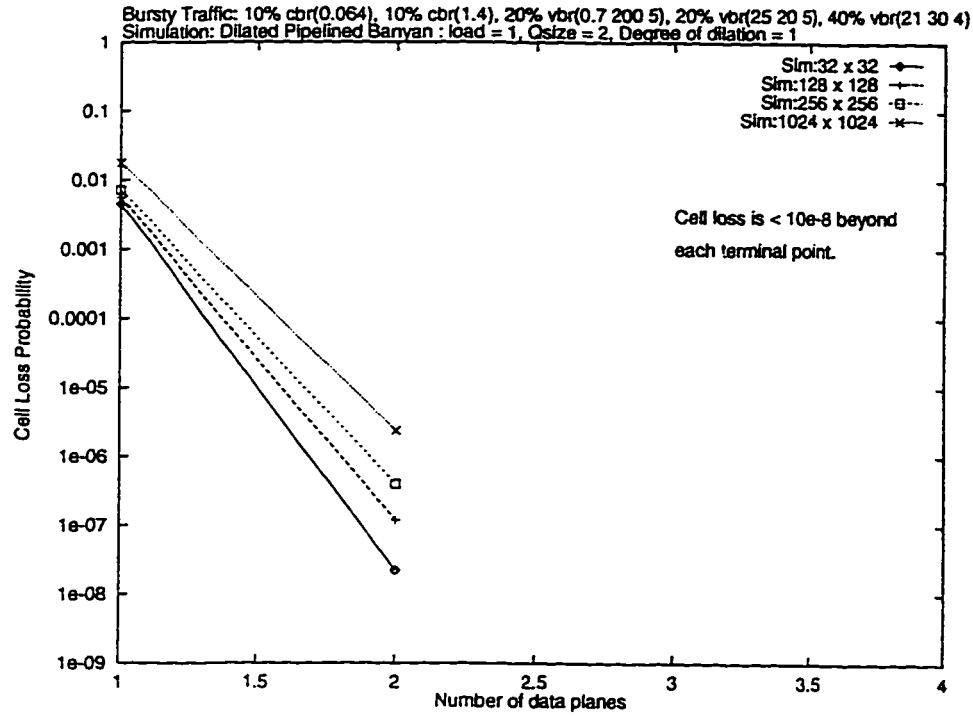

Figure 5.4: Performance of pipelined dilated banyan under ATM traffic mix-1.

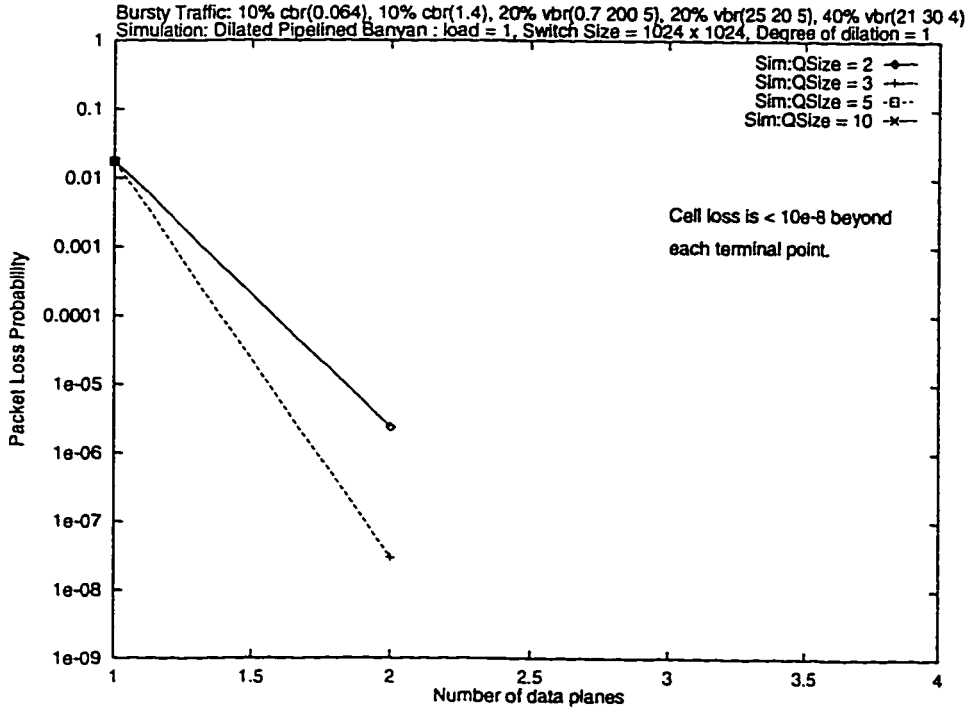

Figure 5.5: Effect of buffering on a pipelined dilated banyan under ATM traffic mix-1.

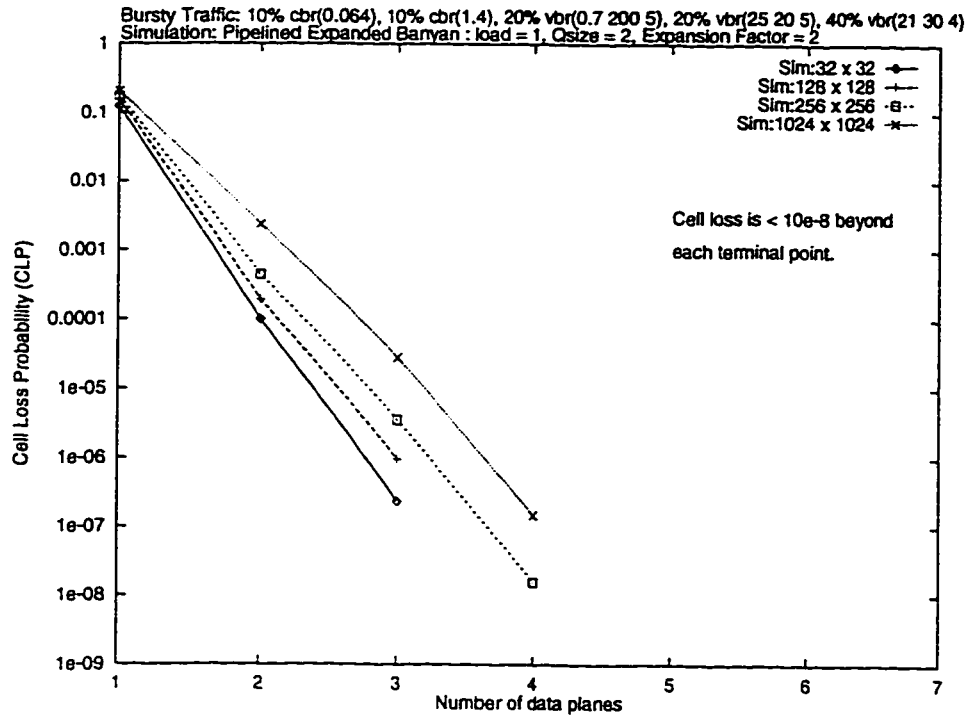

Figure 5.6: Performance of pipelined expanded banyan(EF=2) switches under ATM traffic mix-1.

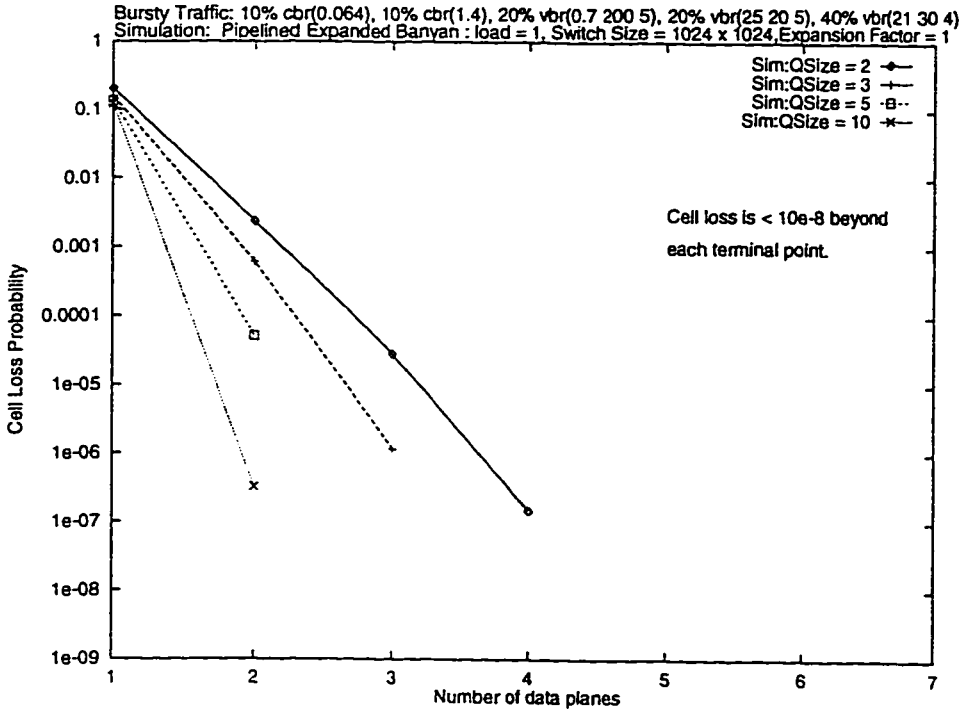

Figure 5.7: Effect of varying buffer size on a pipelined expanded banyan(EF=2) Switch under ATM traffic mix-1.

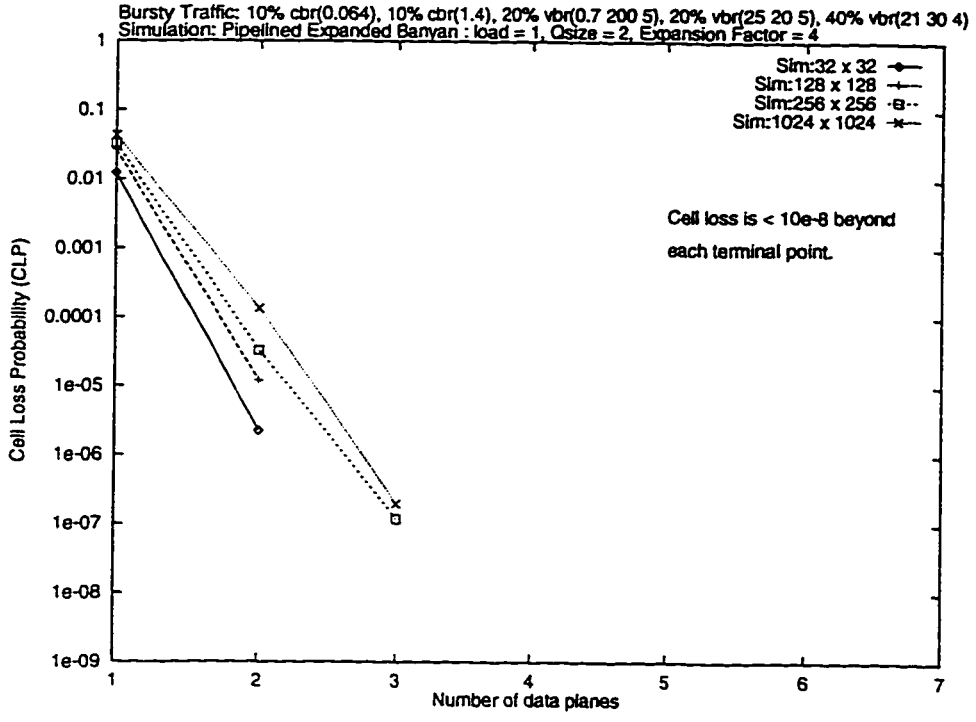

Figure 5.8: Performance of pipelined expanded banyan(EF=4) switches under ATM traffic mix-1.

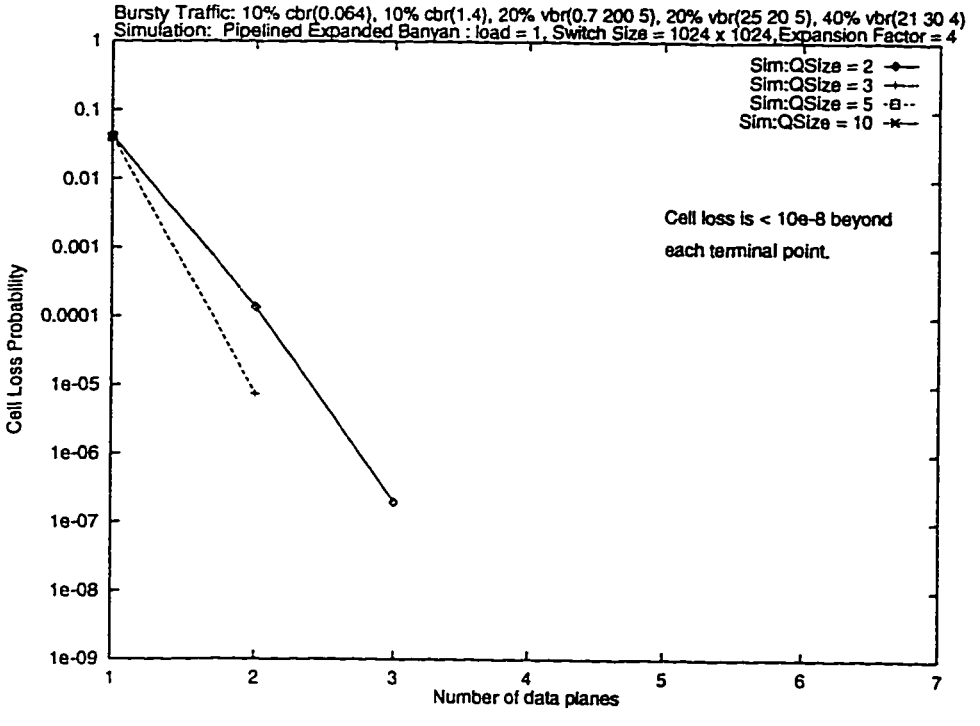

Figure 5.9: Effect of varying buffer size on a pipelined expanded banyan(EF=4) Switch under ATM traffic mix-1.

5.3.2 Performance under Traffic Mix 2.

In this section we present the simulation results of different pipelined switches when subjected to ATM Traffic mix-2 which is described in previous section. Traffic mix-2 has a lesser number of VBR sources than the Traffic mix-1. Therefore this type of traffic mix subjects the switch to a load less than the traffic mix-1. This fact can be observed by comparing the following figures with the one in the previous section. Here we have presented the simulation results of only pipelined simple banyan and pipelined dilated banyan. For the purpose of comparison, similar simulation results have been presented for each type of pipelined switch. For example the pipelined simple banyan results under traffic mix-2 shown in Figures 5.10 and 5.11 correspond to the results shown in Figures 5.2 and 5.3. We can see that the number of data planes required to achieve a cell loss around 10^{-7} reduces from 6 in traffic mix-1 to 4 in traffic mix-2. Similarly the results of pipelined dilated banyan presented in Figures 5.2 and 5.3 show a reduction in the number of data planes required under traffic mix-2 when compared to the results of similar switches under traffic mix-1.

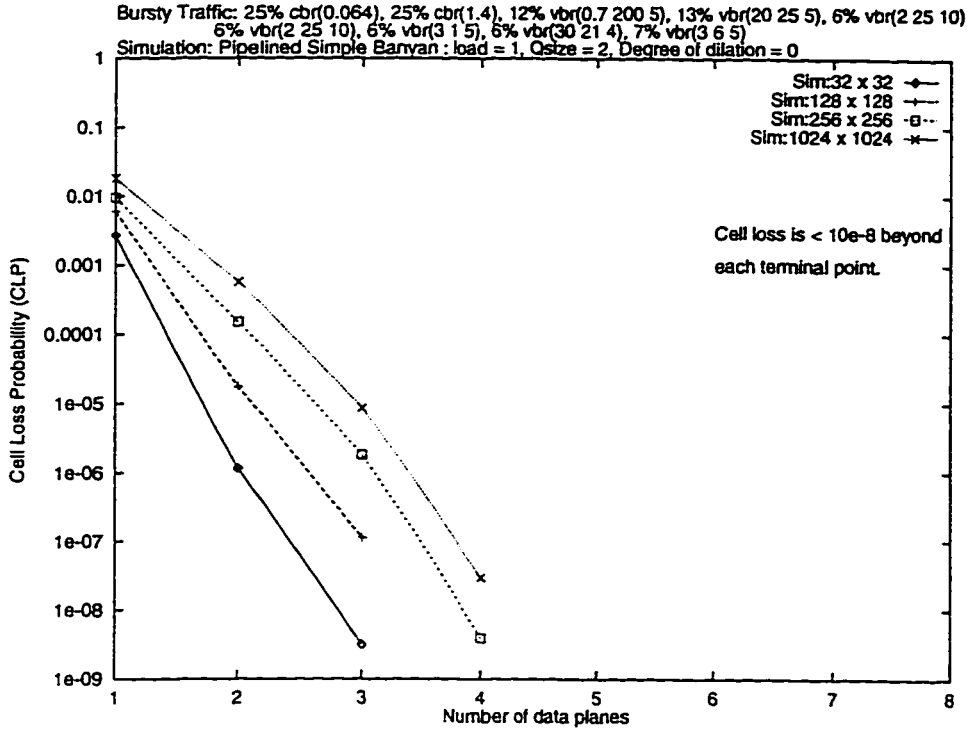

Figure 5.10: Performance of pipelined simple banyans under ATM traffic mix-2.

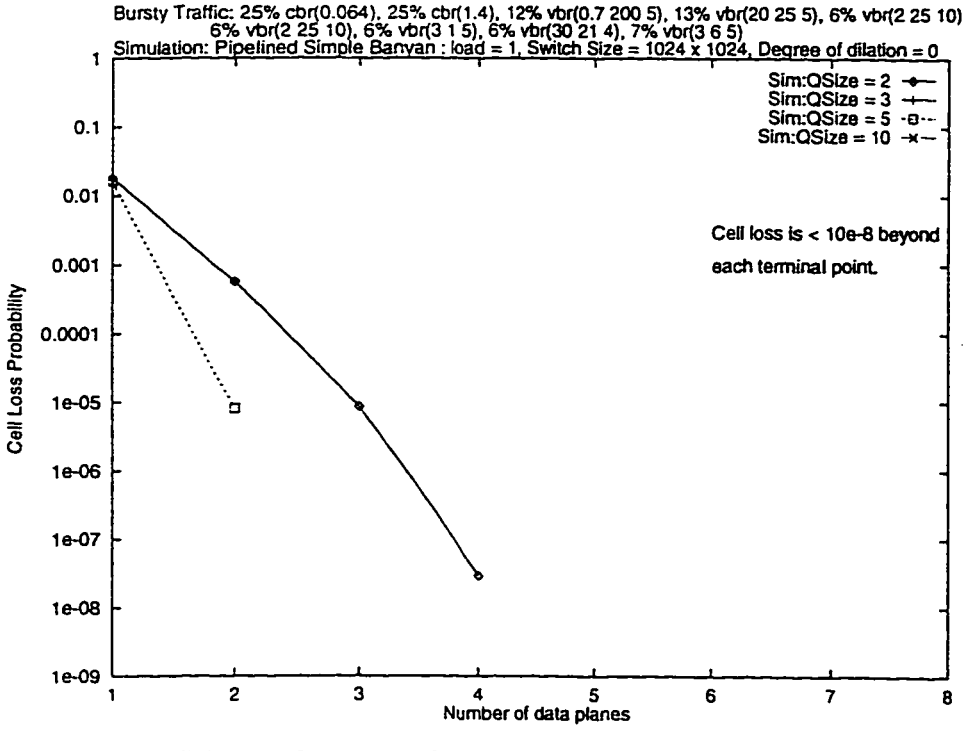

Figure 5.11: Effect of varying buffer size on a pipelined simple banyan under ATM traffic mix-2.

Bursty Traffic: 25% cbr(0.064), 25% cbr(1.4), 12% vbr(0.7 200 5), 13% vbr(20 25 5), 6% vbr(2 25 10) 6% vbr(2 25 10), 6% vbr(30 21 4), 7% vbr(3 6 5)

Simulation: Dilated Pipelined Banyan : load = 1, QSize = 2, Degree of dilation = 1

Switch Size	Dataplanes	Cell Loss Probability
32×32	1	9.125000e-05
32×32	2	$\leq 10^{-8}$
128×128	1	5.324297e-04
128×128	2	$\leq 10^{-8}$
256×256	1	2.969961e-04
256×256	2	$\leq 10^{-8}$
1024×1024	1	1.036016e-03
1024×1024	2	$\leq 10^{-8}$

Table 5.2: Performance of a pipelined dilated banyans under ATM traffic mix-2.

Bursty Traffic: 25% cbr(0.064), 25% cbr(1.4), 12% vbr(0.7 200 5), 13% vbr(20 25 5), 6% vbr(2 25 10) 6% vbr(2 25 10), 6% vbr(30 21 4), 7% vbr(3 6 5)

Simulation: Dilated Pipelined Banyan : load = 1, Switch Size = 1024×1024, Degree of dilation = 1

Queue Size	Dataplanes	Cell Loss Probability
2	1	1.036016e-03
2	2	$\leq 10^{-8}$
3	1	1.032568e-03
3	2	$\leq 10^{-8}$
5	1	1.038418e-03
5	2	$\leq 10^{-8}$
10	1	1.035176e-03
10	2	$\leq 10^{-8}$

Table 5.3: Effect of varying buffer size on a pipelined dilated banyan under ATM traffic mix-2.

5.4 Conclusion

In this chapter we presented the simulation results of four different pipelined switches namely pipelined simple banyan, pipelined dilated banyan with dilation degree 1, pipelined expanded banyan with expansion factor 2 and pipelined expanded banyan with expansion factor 4. The simulations were performed for two different types of ATM traffic sources. In the simulator, ON/OFF source model has been used to generate traffic for VBR sources. This model requires three parameters to be specified for each type of VBR source, the mean cell arrival rate m, the maximum burst length B, and the burstiness factor β. A traffic mix which has more VBR sources subjects the switch to a higher input load. The reason is that the peak cell rate of a typical VBR is much higher than the cell rate of a CBR source. This was verified from the simulation results which show that, to achieve a particular level of performance, lesser hardware resources were required under traffic mix-2 than under traffic mix-1 since traffic mix-2 has less number of VBR sources.

Chapter 6

Conclusions

Computer communication networks have evolved from centralized mainframe computing into the era of the distributed processing client-server environment. The first communications network resembled a hierarchical or star topology. Whereas today there are a number of heterogeneous networks with different topologies. These heterogeneous networks are interconnected together and provide a distributed processing environment so that several client workstations can communicate with distributed servers to extract their core information. Applications using this type of environment such as WWW, video conferencing and telemedicine put an increasing demand, not only on the transmission bandwidth, but also on the integration of various types of services like voice, video, and data. Traditional networks are specialized networks supporting only a specific type of service. Circuit switched networks provide voice services and packet switched networks provide data exchange services. N-ISDN (narrowband ISDN) which is basically a digital circuit switched technology alleviated the situation by integrating the voice and non-voice services. However it could not sustain the high bandwidth requirements of the present multimedia appli-

cations accessing Internet. B-ISDN solves the problem by adopting ATM, a transfer technology that scales up to any bandwidth [29].

B-ISDN service is actually a compromise between pure circuit switching and pure packet switching. The actual service offered is connection oriented but it is implemented internally with packet switching, not circuit switching. The transmission technique adopted for B-ISDN is ATM (Asynchronous transfer mode) which is a fixed size cell-switching technology. This type of cell switching is very flexible and can handle both constant rate traffic (audio, video) and variable rate traffic (data) easily.

Previously the bottleneck in B-ISDN was transmission links. However with the advent of fibre optic links this has shifted to processing speeds at the switching nodes and the propagation delay of the channel [8]. In this thesis we took-up the task of investigating the various switch structures in an attempt to come up with a new design that gives better throughput. We have basically worked on some arrangement of banyan networks and made full use of parallelization in order to achieve high throughput and low switching delay. We started by giving a brief introduction to ATM definition and concepts in the first chapter.

In chapter 2 we have presented an extensive literature survey of various switch architectures proposed to date. A classification of switch architectures is also given. From the survey, we found that most of the research on switch architectures emphasizes on designing multiple outlet space division switches. Multiple outlet switches can be obtained primarily by two techniques, replication or dilation. In replication a number of banyans are replicated either in horizontal direction (TBSF) or vertical direction (MBSF) and in dilation technique internal links are increased to improve

the throughput of the switch.

In chapter 3 we have discussed the design issues of pipelined dilated banyan switch which is based on pipelined simple banyan proposed by Wong and Yeung [25]. We have used a dilated banyan in each data plane to improve the throughput of each data plane. Pipelined dilated banyan has three important features: link dilation, pipelining, and input buffering. In chapter 3 we have also presented equations to evaluate the hardware resources required in pipelined dilated banyan and pipelined expanded banyan.

In chapter 4 we have presented the analytical models under uniform traffic for dilated banyan, pipelined simple banyan and pipelined dilated banyans. The simulation results for these switches have been compared with the analytical results and the effect of correlation in the output port selection has been investigated. Pipelined dilated banyan was found to use a lesser number of data planes than pipelined simple banyan for a particular level of performance. The complexity of dilated banyan is high. Therefore we have presented the simulation results of pipelined expanded banyan which has less complex expanded banyans in each data plane. We have presented the simulation results of pipelined expanded banyan with expansion factor 2 and pipelined expanded banyan with expansion factor 4. The results of pipelined expanded banyan with expansion factor 4 are similar to pipelined dilated banyan with dilation degree 1. The hardware resources required in pipelined expanded banyan with expansion factor 4 is slightly less than pipelined dilated banyan. Since the delay is also less, we can conclude that pipelined expanded banyan with expansion factor 4 performs better than pipelined dilated banyan.

In chapter 5 we have presented the simulation results under ATM traffic for four dif-

ferent types of pipelined switches namely pipelined simple banyan, pipelined dilated banyan with dilation degree 1, pipelined expanded banyan with expansion factor 2, and pipelined expanded banyan with expansion factor 4. We have performed simulations under two types of ATM traffic mixes and observed the variation in the load of the input traffic due to variation in the number of VBR traffic sources.

References

[1] Hamid Ahmadi and Wolfgang E.Denzel. A survey of modern high-performance switching techniques. *IEEE Journal on selected areas in communications*. 7(7):1091–1103, september 1989.

[2] A.Huang and S.Knauer. Starlite : A wideband digital switch. *Proceedings GLOBECOM 84, Atlanta. GA*. (12):121–125, December 1984.

[3] Mayez Al-Mohammed and Lubomir Bic. Combining Linear Data Patterns for Accessing Parallel Memories Through Arbitrary Multistage Networks. *IEEE Transactions on Parallel and Distributed Systems*, September 1995.

[4] Mayez Al-Mohammed and S. Seiden. Minimization of Memory and Network Contention for Accessing Arbitrary Data Patterns in SIMD Systems. *IEEE Transactions on Computers*, 45(6):757–762, June 1996.

[5] Mayez Al-Mohammed and S. Seiden. A Heuristic Storage for Minimizing Access Time of Arbitrary Data Patterns. *IEEE Transactions on Parallel and Distributed Systems*. 8(4):441–447, April 1997.

[6] Mayez Al-Mouhamed, Habib Youssef, and Wasif Hasan. A Novel Fast Parallel Tree Banyan Switch Fabric. *Submitted to International Journal on Computer Systems*, 1996.

[7] Ra'ed Y. Awdeh and H.T. Mouftah. The expanded delta fast packet switch. *IEEE transcations on Computers*, 1994.

[8] J.J. Bae and T. Suda. Survey of Traffic control schemes and protocols in ATM networks. *Proceedings IEEE*, 79(2):170–189, February 1991.

[9] Jonathan Chao and Dipak Ghosal. Connectionless Service for Public ATM Networks. *IEEE Communications*, 32(8):34–42, Aug 1994.

[10] Andrew Day. International Standardization of BISDN. *IEEE Spectrum*, 6(6):143–150, june 1991.

[11] Martin de Prycker. *Asynchronous Transfer Mode Networks : Solutions for broadband ISDN*. Ellis Horwood, 1991.

[12] Harry J.R. Dutton and Peter Lenhard. *Asynchronous Transfer Mode - Technical Overview*. Prentice Hall, 2nd edition, 1995.

[13] J.Giacopelli et al. Sunshine : A high performance self-routing broadband packet-switch architecture. *IEEE journal on selected areas in communications*, (10):1289–1298, october 1991.

[14] Jeff Gould. ATM's long strange trip to the mainstream. *Data Communications*, 6(6):120–130, june 1994.

[15] Toshihiro Hanawa, Hideharu Amano, and Yoshifumi Fujikawa. Multistage Interconnection Networks with multiple outlets. *International Conference on Parallel Processing*. I-1 to I-8(8):1173–1192, october 1994.

[16] J.J.Li and C.M.Wang. B-Tree : A high performance fault ATM switch. *IEE - Proc-communications*, 141(1):20–28, february 1994.

[17] K.E.Batcher. Sorting Networks and their applications. *AFIPS Proceedings 1968 Spring Joint Computer Conference*, 32(10):307–314, october 1968.

[18] James lane. ATM knits voice, data on any net. *IEEE Spectrum*, 13(2):42–45, February 1994.

[19] Joseph L.Hammond and Peter.O'Really. *Performance Analysis of Local Computer Networks*. chapter 2, pages 111–123. Addison-Wesley Publishing Company, 1988.

[20] Donald R. Marks. ATM from A to Z. *Data Communications*, 12(12):26–29, december 1994.

[21] David E. McDysan and Darren L. Spohn. *ATM Theory and Applications*, chapter 1, pages 30–32. McGraw Hill International Editions Hall, 1995.

[22] Arthur Miller. From here to ATM. *IEEE Spectrum*, 6(6):20–24, june 1994.

[23] M.J.Narasimha. Batcher-Banyan Self-routing network : universality and simplification. *IEEE Transactions on communications*, 36(10):1175–1178, october 1988.

[24] Raif O.Onvural. *Asynchronous Transfer Mode Networks : performance Issues*, chapter 13. Rockville, Md Computer Scince Press, 2nd edition, 1983.

[25] P.C.Wong and M.S. Yeung. Design and analysis of a Novel Fast Packet Switch - Pipeline Banyan. *IEEE/ACM Transcations on Networking*, 3(1):63–69, february 1995.

[26] Reza Rooholamin and Vladimir Cherkassky. Finding right ATM switch for the market. *IEEE Computer*, 27(4):16–28, april 1994.

[27] Debanjan Saha and Satish K. Tripati. IP on ATM Local Area Networks. *IEEE Communications*, 32(8):52–59, Aug 1994.

[28] G D Stamoulis, M E Anagnostou, and A D Georgantas. Traffic source models for ATM networks: a survey. *computer communications*, 17(6), june 1994.

[29] Andrew S. Tanenbaum. *Computer Networks*, chapter 1,3, pages 61–66,139–155. Prentice Hall, third edition, 1996.

[30] Fouad A. Tobagi. Fast Packet Architectures for Broadband Integrated services. *Proceedings of the IEEE*, 78(1):133–166, january 1990.

[31] Fouad A. Tobagi, Timothy Kwok, and Fabio M.chiussi. Architecture, Performance, and Implementation of Tandem Banyan Fast Packet Switch. *IEEE Journal on selected areas in communications*, 9(8):1173–1192, october 1991.

[32] Hong Linh Truong, William W. Ellington Jr., Jean Yues Le Boudec, Andreas X. Meier, and J. Wayne Pace. LAN Emulation on an ATM Network. *IEEE Communications*, 33(5):70–85, May 1995.

[33] Indra widjaja and Alberto Leon Garcia. The Helical Switch: A multi-path ATM switch which preserves cell sequence. *IEEE Transcations on Communications*, 42(8), august 1994.

[34] Ra'ed Y.Awdeh and H.T.Mouftah. Survey of ATM switch architectures. *Computer Networks and ISDN Systems*, 7(7):1567–1613, september 1995.

[35] Y.Yeh, M.Hluchyj, and A.Acampora. The Knockout Switch : A simple, modular architecture for high performance packet-switching. *IEEE journal on selected areas in communications*, 5(8):1274–1283, october 1987.

[36] Ellen Witte Zegura. Architecture for ATM switching systems. *IEEE communications magazine*, (2):28–37, february 1993.

Vita

- Mohammed Kaleemuddin
- Born in Hyderabad. India on December 20, 1970
- Received Bachelor's degree in Computer Science and Engineering from Osmania University Hyderabad. India in July , 1992.
- Completed Master's degree requirements at King Fahd University of Petroleum and Minerals, Dhahran. Saudi Arabia in June, 1997.